

Iteración 1:

Elementos relacionados con el dominio:

- Proyecto (information holder)
- Reporte (information holder)
- Participante (information holder)

Elementos relacionados con el sistema:

- Aplicación (interface)

Cosas que fluyen dentro del sistema:

- Actividad (information holder)

Responsabilidades:

- Iniciar un nuevo proyecto
- Ver los proyectos creados
- Añadir actividades a un proyecto
- Añadir nuevos participantes a un proyecto.
- Cronometrar el trabajo
- Hacer actividad
- Modificar fecha/hora de la actividad
- Cambiar persona que hizo actividad
- Escoger un proyecto específico
- Escoger una actividad específica
- Visualizar el tiempo que tardó una actividad en realizarse.
- Medir cuánto tarda en realizarse una actividad
- Generar un reporte
- Visualizar el reporte
- Calcular la información del reporte
- Clasificar la actividad por tipo.
- Tener persistencia

Iteración 2:

Elementos relacionados con el dominio:

- Proyecto (structurer)
- Reporte (service provider)
- Participante (information holder)

Elementos relacionados con el sistema:

- Aplicación (interface)

Cosas que fluyen dentro del sistema:

- Actividad (information holder)

Responsabilidades:

- Iniciar un nuevo proyecto
- Ver los proyectos creados
- Añadir actividades a un proyecto
- Añadir nuevos participantes a un proyecto.
- Cronometrar el trabajo
- Introducir información de la actividad
- Iniciar la actividad
- Finalizar la actividad
- Modificar fecha/hora de la actividad
- Cambiar persona que hizo actividad
- Escoger un proyecto específico
- Escoger una actividad específica
- Visualizar el tiempo que tardó una actividad en realizarse.
- Medir cuánto tarda en realizarse una actividad
- Calcular el tiempo total de una actividad
- Generar un reporte
- Visualizar el reporte
- Calcular la información del reporte
- Clasificar la actividad por tipo.
- Tener persistencia

Colaboraciones

- Aplicación, Proyecto, Actividad y Participante: Para poder hacer una actividad necesitan colaborar aplicación, proyecto, actividad y participante. Aplicación tiene o crea un proyecto, en el cual proyecto crea una actividad con su información correspondiente y finalmente al acabar se le asigna al participante que la realizo.
- Aplicación, Proyecto, Actividades, Participante y Reporte: Para producir y mostrar un reporte se necesita que Participante tenga la información de sus actividades (Actividad), luego Proyecto coge esa información y genera un reporte con ayuda de la clase Reporte y finalmente Aplicación coge el reporte y lo muestra.
- Aplicación, Proyecto y Participante: Para añadir participantes a los proyectos. Aplicación debe tener un proyecto actual, luego proyecto añade a un participante con su información de la clase.

Iteración Final

Elementos relacionados con el dominio:

- Proyecto (structurer)
- Reporte (service provider)
- Participante (information holder)

Elementos relacionados con el sistema:

- Aplicación (coordinator)

Cosas que fluyen dentro del sistema:

- Actividad (information holder)

Responsabilidades:

- Iniciar un nuevo proyecto
- Ver los proyectos creados
- Añadir actividades a un proyecto
- Añadir nuevos participantes a un proyecto.
- Cronometrar el trabajo
- Introducir información de la actividad
- Iniciar la actividad
- Finalizar la actividad
- Modificar fecha/hora de la actividad
- Cambiar persona que hizo actividad
- Escoger un proyecto específico
- Escoger una actividad específica
- Visualizar el tiempo que tardó una actividad en realizarse.
- Medir cuánto tarda en realizarse una actividad
- Calcular el tiempo total de una actividad
- Generar un reporte
- Visualizar el reporte
- Calcular la información del reporte
- Clasificar la actividad por tipo.
- Tener persistencia
- Guardar archivos planos
- Leer archivos planos

Colaboraciones

- Aplicación, Proyecto, Actividad y Participante: Para poder hacer una actividad necesitan colaborar aplicación, proyecto, actividad y participante. Aplicación tiene o crea un proyecto, en el cual proyecto crea una actividad con su

información correspondiente y finalmente al acabar se le asigna al participante que la realice.

- Aplicación, Proyecto, Actividad: Para poder modificar una actividad necesitan colaborar aplicación, proyecto y actividad. Aplicación tiene el proyecto, en el cual proyecto tiene una actividad la cual toca modificar según su información.
- Aplicación, Proyecto, Actividades, Participante y Reporte: Para producir y mostrar un reporte se necesita que Participante tenga la información de sus actividades (Actividad), luego Proyecto coge esa información y genera un reporte con ayuda de la clase Reporte y finalmente Aplicación coge el reporte y lo muestra.
- Aplicación, Proyecto y Participante: Para añadir participantes a los proyectos. Aplicación debe tener un proyecto actual, luego proyecto añade a un participante con su información de la clase.
- Aplicación y Proyecto: Para crear un proyecto, aplicación crea el proyecto, con su información correspondiente de la clase.

Reflexion:

Una reflexión sobre el diseño obtenido:

- Un trade-off, es tener un estilo de control centralizado, ya que por una parte este nos permite tener todo controlado desde la clase Proyecto, pero en un futuro si crece o cambia la aplicación se puede dificultar que Proyecto maneje tantos métodos y conozca tanta información.
- Una decisión de diseño que tomamos fue que participante no fuera el que modificara las actividades, sino que proyecto fuera el que lo hiciera, esto para evitar relaciones innecesarias y preservar la cohesión.
- Otra decisión de diseño fue no tener una clase externa dueño, ya que complicaba las cosas innecesariamente, sino que tener que los participantes tengan un booleano que defina si es dueño o no y actuara consecuente a esto

Reflexión sobre el proceso de diseño:

- Un elemento que presento dificultad fue decidir si debiese existir la clase dueño y cual debería de ser su relación con participante y actividades. Al final decidimos no implementarla para evitar enredos y solamente incorporarla a Participante como un booleano
- Otro elemento que presento gran análisis fue la clase Proyecto, esto debido a que al inicio no sabíamos que responsabilidades darle o que propósito ponerle. Solo que contenga información del proyecto o que realice acciones sobre él y la actividad. Al final nos decantamos porque proyecto fuera nuestra clase central ya que tenía que conocer la información de casi todos y era practico de una vez asignarle los métodos correspondientes.

- Finalmente, Reporte, presentó una leve dificultad debido a que no sabíamos con qué relacionarla o si debería de saber alguna información. Al final decidimos que como ya teníamos la clase central de proyecto, que esta se encargara de hacer el reporte, debido a que ya tenía la información disponible y que reporte fuera un service provider que se permitiera construir y obtener.