**Documento de Diseño Proyecto**

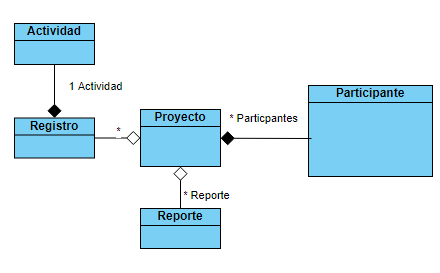
1. **Iteración**
   1. **Roles**
2. Aplicación<<controller>>

Juego

* 1. **Responsabilidades**

|  |  |
| --- | --- |
| Responsabilidad | Rol |
| Correr aplicación | aplicación |

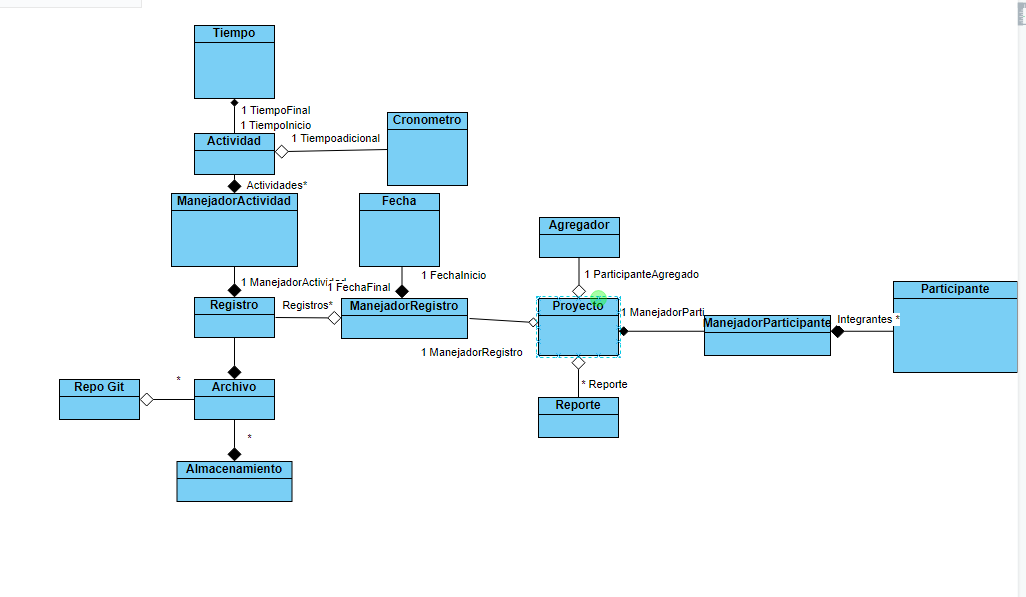
1. **Iteración**
   1. **Roles**
2. Proyecto<<estructurer>>
3. Participante<<information holder>>
4. Actividad<<structurer>>
5. Registro<<info holder>>
6. Reporte<<structurer>>



* 1. **Responsabilidades**

|  |  |
| --- | --- |
| Responsabilidad | Rol |
| Crear proyecto | proyecto |
| Agregar particpante | particpante |
| Registrar actividad | participante |
| Mostrar reporte | reporte |
| Tomar tiempo | actividad |
| Modificar datos | participante |
| Mirar reporte | participante |

1. **Iteración** 
   1. **Roles**
      1. Proyecto<<estructurer>>
      2. Participante<<information holder>>
      3. Actividad<<structurer>>
      4. Registro<<info holder>>
      5. Reporte<<structurer>>
      6. Repo git/service
2. Archivo/info holder
3. Almacenamiento/actividades
4. Fecha/info
5. Cronometro/ service
6. Agregador participante/ strcuturer
7. Tiempo/ info holder
8. Manejadoractividades/structurer
9. Mnaejadoregistro/strcuturer
10. Manejadorparticpantes/strcutrer



* 1. **Responsabilidades**

|  |  |
| --- | --- |
| Responsabilidad | Rol |
| Crear proyecto | proyecto |
| Agregar particpante | particpante |
| Registrar actividad | participante |
| Mostrar reporte | reporte |
| Tomar tiempo | actividad |
| Modificar datos | participante |
| Mirar reporte | participante |
| Manejar versiones | Repo git |
| Escribir actividad | archivo |
| Almacenar actividad | almacenamiento actividades |
| Registrar fecha inicio | fecha |
| Registrar fecha final | fecha |
| Pausar tiempo | cronometro |
| Agregar particpante | agregador psrticpante |
| Tomar tiempo activdad | tiempo |
| Verificar info actividades | manejador actvidades |
| Verficiar registros | manejador registro |
| Mirar particpantes | manejador particpantes |

1. **Iteración**
2. **Roles**
   * 1. Proyecto<<estructurer>>
     2. Participante<<information holder>>
     3. Actividad<<structurer>>
     4. Registro<<info holder>>
     5. Reporte<<structurer>>
     6. Repo git/service
3. Archivo/info holder
4. Almacenamiento/actividades
5. Fecha/info
6. Cronometro/ service
7. Agregador participante/ strcuturer
8. Tiempo/ info holder
9. Manejadoractividades/structurer
10. Mnaejadoregistro/strcuturer
11. Manejadorparticpantes/strcutrer
12. Manejador version/ strcuturer
13. Registraractividad/ info holder
14. TIpoactividad/ info holder
15. Manejadorarchivos/ strcuturer
16. Participante inicial/ info holder
17. **Responsabilidades**

|  |  |
| --- | --- |
| Responsabilidad | Rol |
| Crear proyecto | proyecto |
| Agregar particpante | particpante |
| Registrar actividad | participante |
| Mostrar reporte | reporte |
| Tomar tiempo | actividad |
| Modificar datos | participante |
| Mirar reporte | participante |
| Manejar versiones | Repo git |
| Escribir actividad | archivo |
| Almacenar actividad | almacenamiento actividades |
| Registrar fecha inicio | fecha |
| Registrar fecha final | fecha |
| Pausar tiempo | cronometro |
| Agregar particpante | agregador psrticpante |
| Tomar tiempo activdad | tiempo |
| Verificar info actividades | manejador actvidades |
| Verficiar registros | manejador registro |
| Mirar particpantes | manejador particpantes |
| Verificar versiones | manejador versiones |
| Registrar nombre | registrar actividad |
| Definie tipo | tipoactividad |
| Veficiar archivo plano | manejador archivos |
| Crear proyecto | particpante inicial |

**El diagrama final está en el pdf del repo Entrega 2**

**GLOSARIO**

1. Manejador versión: Esta clase se va a encargar de manejar las versiones que se van a mandar al repositorio. En este caso, verificaría que sean archivos de texto planos.
2. Registrar actividad: Esta clase hace referencia a la que se va a encargar de registrar el usuario y la actividad que se este realizando en ese momento.
3. Tipo actividad: Esta clase hace referencia a la hora de definir los tipos de cada actividad.
4. Manejador archivos: Esta clase como su nombre lo indica se va a encargar de verificar los archivos que de van a almacenar de registros.
5. Participante inicial: Esta clase va a ser una herencia, de participante. Y su función es ser el dueño del proyecto, es decir, de poder agregar participantes.
6. Repo git/service: Esta clase hace referencia al servicio externo del proyecto. En el cual se van a almacenar las versiones del mismo.
7. Archivo: Esta clase es un texto el cual es el que se va a almacenar y a subir para tener el registro.
8. Almacenamiento: Esta clase se va a encargar de almacenar los archivos en el dispositivo, en el cual se esta trabajando.
9. Fecha: Esta clase es una extensión para predefinir las fechas y tener un formato cuando se pida.
10. Cronometro: Esta clase hace referencia al cronometro que se va llevar y mostrar al usuario para que este pause o reinicie su medida en la actividad.
11. Agregado participante: Esta clase hace referencia a lo que se necesita para poder agregar un participante. Es decir, nombre y correo y soltar un participante.
12. Tiempo: Esta es una clase la cual hace referencia al tiempo promedio y los diferentes tiempos que se le dan al usuario por actividad y por reporte.
13. Manejador actividades: Esta clase se va a encargar de administrar todas las actividades, verificar si se hicieron actividades en varios días por decir un ejemplo.
14. Manejador registro: Esta clase se encarga de administrar los registros. Por ejemplo: organizarlos, o ordenarlos.
15. Manejador participante: Esta clase es la que se encarga de llevar una administración de todos los participantes que están el proyecto.
16. Proyecto: Es el contender o la pieza principal de nuestra aplicación, en el van a estar todas las interacciones más importantes.
17. Participante: Esta clase hace referencia a los objetos que van a participar en el proyecto y puede ser tomada como interface.
18. Actividad: Esta clase va a ser una unión de varis y va a hacer la que haga la interacción con el sistema.
19. Registro: Esta clase hace referencia a la información que se va a guardar en los archivos y registros. Además, de ser usada para estadísticas internas
20. Reporte: Es el registro que se genera de cada actividad y es el que se va manejar en otros lados. Además, de tener la posibilidad de visualizarlo.