Trabajo Practico Final

Ciencia Participativa y Juegos

Integrantes:

- Souto Maximiliano
- Giraudo Matías
- Carrizo Nicolás

Patrones de diseño:

State:

Para los estados en los que puede estar un desafío del usuario utilizamos el patrón State. Los roles que representan son:

- **Context:** DesafioUsuario
- **State:** IEstadoDesafio
- **ConcreteState**: EstadioSinIniciar, EstadoAceptado, EstadoVencido y EstadoCompletado

Strategy:

Para las recomendaciones de desafíos que se le puede dar a un Usuario utilizamos el patrón Strategy, este patron nos va a permitir recomendarles desafíos a nuestros Usuarios, se encuentran dos formas de recomendacion RecomendadorPorPreferencia o RecomendadorPorFavorito. Los roles que representan son:

- **Context:** Usuario
- **Strategy:** IRecomendador
- **ConcreteStrategy:** RecomendadorPorCoincidencia y RecomendadorPorFavorito

Composite:

Primero utilizamos el patrón Composite para la parte de búsqueda de proyectos. Lo usamos en esta parte para poder hacer búsquedas combinadas por titulares o categorías. Los roles que representan son:

- **Component:** BuscadorProyectos
- **Composite:** BusquedaAvanzada
- **Leaf:** BusquedaPorCategoria, BusquedaPorTitulo, BuscadorAnd, BuscadorOr y BuscadorNot

Segundo utilizamos el patron Composite para las restricciones temporales que van a tener nuestros Desafíos. Los roles que representan son:

- **Component:** IRestriccionTemporal
- **Composite:** RestriccionCombinada
- **Leaf:** RestriccionFinSemana, RestriccionSemana y RestriccionFecha

Decisiones de diseño tomadas:

- Nuestra clase Usuario tiene una lista de DesafioUsuario, esta clase representa un desafío en particular para un Usuario. Esta clase es la que va a manejar los estados actuales del desafío que tiene ese Usuario.
- Utilizamos ENUM pare representar la Dificultad de un Desafío. En donde MUYFACIL es representado por un 1 y MUYDIFICIL con un 5. También utilizamos enum para los días de la semana. Donde LUNES es 1 y DOMINGO es 7.
- Para los test utilizamos mock para mockear clases que no son la clase principal a testear. Los test están divididos en tres partes. Tenemos un setup, exercise y verify. En el README dejamos los jar utilizados.
- Representamos una clase Sistema esta clase representa el sistema de nuestra aplicación. Su objetivo es almacenar todos los desafíos, proyectos, usuarios y categorías que van a ser utilizados en la aplicación.
- Tenemos una clase Coordenada que representamos los puntos x e y con el tipo Integer.