

**UNIVERSIDAD TECNOLÓGICA NACIONAL  
FACULTAD REGIONAL RESISTENCIA**

*Ingeniería en Sistemas de Información*

**PARADIGMAS DE LA PROGRAMACIÓN**

**Profesores:** Bernal, Rubén Alfredo

Cuevas, Carlos

**Alumnos:** Salas, Nahíara Juliana

Vázquez, Máximo Ezequiel

Duarte, Hugo Emmanuel

Aquino, Marcos J. Alejandro

## Resumen

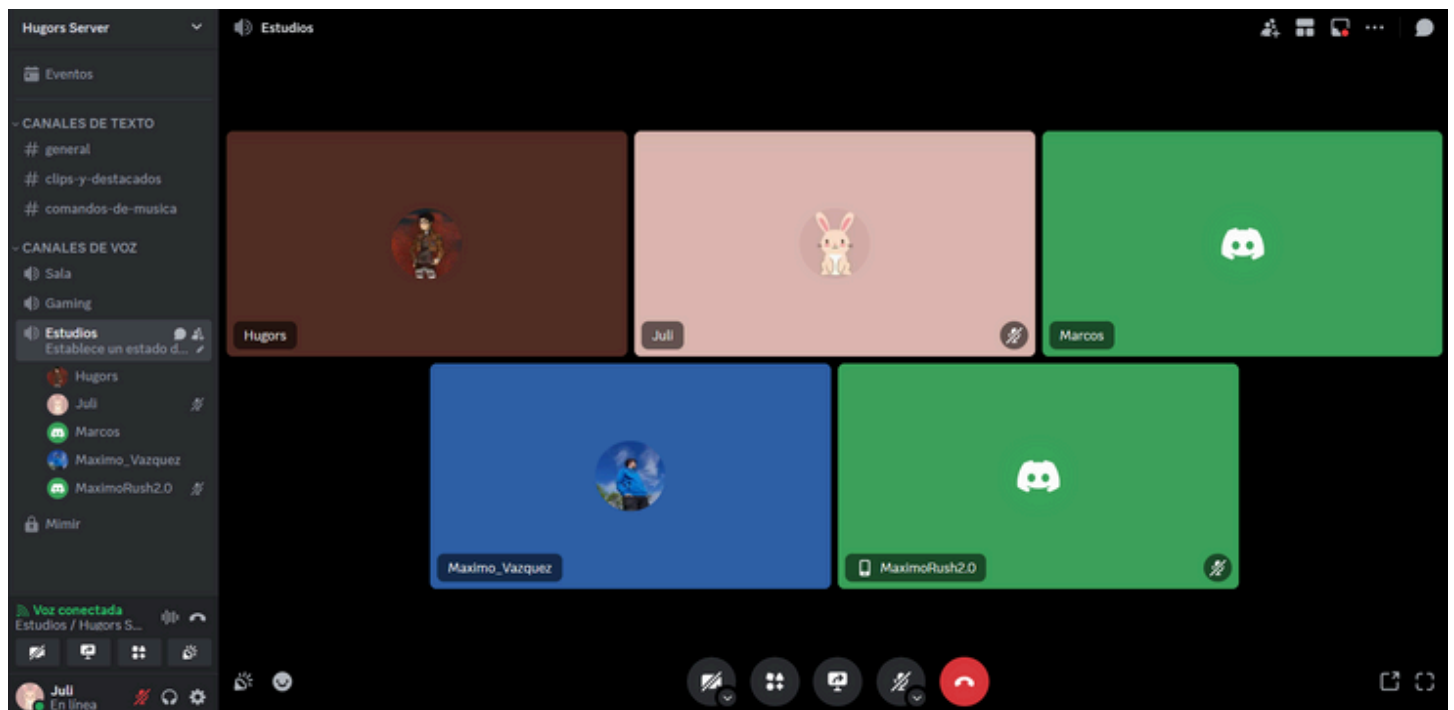
En el presente trabajo se llevará a cabo la preparación, planificación y codificación de un juego llamado “The Floor: La Conquista”. Se llevará una bitácora detallando los días que se realicen avances en el proyecto y se detalla cuáles serán los avances llevados a cabo. Así mismo se especificarán las partes del juego y su desarrollo. El juego consta de un tablero dividido en casilleros los cuales, cada uno de ellos les pertenece a uno de los jugadores, el objetivo del juego es conseguir conquistar los casilleros de los demás jugadores, el ganador será aquel jugador que conquiste todos los casilleros del tablero, para conquistar el casillero deseado los jugadores dueños de dichos casilleros se enfrentarán en un duelo en donde deberán poner a prueba sus habilidades estratégicas y descubrir si la suerte se encuentra de su lado.

## Introducción

El presente informe detalla la planificación y ejecución del trabajo práctico integrador final, titulado “La conquista”. Este proyecto se basa en un enfoque creativo y estratégico, adaptando las reglas de un juego tradicional a un nuevo contexto diseñado por nuestro equipo. A lo largo del desarrollo, hemos seguido un plan colaborativo que se estructura en varias etapas, destacando la asignación de roles, las decisiones creativas, y el uso de herramientas digitales para una coordinación eficiente.

## Plataforma de Trabajo

Para facilitar la comunicación y la colaboración a distancia, decidimos utilizar la plataforma “Discord” como nuestro principal medio de interacción. Las reuniones fueron realizadas en llamadas grupales, las cuales documentamos mediante capturas de pantalla que serán incluidas como evidencia del trabajo en equipo.



Además, utilizamos un **documento compartido en Word** para trabajar juntos en la redacción del informe, permitiéndonos editar y revisar el contenido de manera simultánea y organizada.

## Bitácora de Reuniones

### Reunión 1: 10/10/2024

En nuestra primera reunión, nos enfocamos en establecer las bases del proyecto:

- **Objetivo:** Alinear nuestras ideas y definir la dirección creativa del trabajo práctico.
- **División de tareas:** Nos organizamos en grupos de dos personas para optimizar el tiempo y el esfuerzo. Cada subgrupo fue asignado a tareas específicas, como la definición de objetos y su funcionamiento, asegurando que se llevará un registro individual de problemas y soluciones encontradas.
- **Inspiración y análisis:** Como equipo, decidimos ver un episodio del juego "The Floor: La conquista" para obtener nuevas ideas y explorar variantes para nuestra propuesta.
- **Investigación técnica:** Analizamos la viabilidad de desarrollar un entorno gráfico en Smalltalk para el juego.
- **Dinámica de conquista:** Discutimos diferentes opciones para la dinámica de conquista de casillas, acordando que cada miembro presentaría una propuesta con sus pros y contras para la siguiente reunión.

### Reunión 2: 17/10/2024

En esta reunión, cada miembro presentó su propuesta, y tras una discusión exhaustiva, evaluamos las ventajas y desventajas de cada idea.

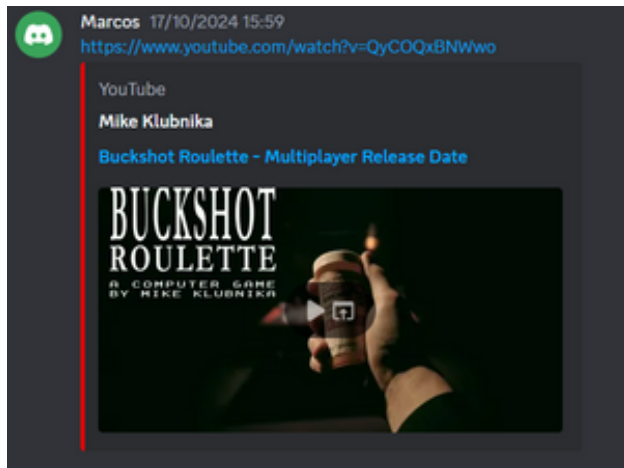
Las propuestas principales fueron:

- **Piedra, papel o Tijera:** El clásico juego de Piedra, Papel o Tijera, pero con un giro estratégico. En lugar de que cada ronda sea aislada, los jugadores seleccionan sus movimientos (piedra, papel o tijera) para una serie de 5 turnos de antemano. Luego, ambos jugadores revelan sus movimientos de manera simultánea cada turno. **Gana** el jugador que gana más rondas tras los 5 turnos. Los jugadores tendrán que intentar predecir los patrones del oponente y hacer movimientos estratégicos anticipando lo que creen que hará el otro.
- **Tateti:** Los jugadores alternan turnos colocando una ficha en un espacio vacío del tablero de 3x3. Después de que cada jugador haya colocado su tercera ficha (es decir, al inicio del cuarto turno de cada jugador), ya no se colocan nuevas fichas. En su lugar, **pueden mover una de sus fichas ya colocadas** a un espacio vacío adyacente (vertical, horizontal o diagonal). El objetivo sigue siendo formar una línea de tres fichas consecutivas, pero ahora las fichas pueden moverse, lo que añade un nivel táctico y permite revertir situaciones donde antes el juego podría haber terminado.
- **Ruleta rusa:** Se toma un revólver imaginario con 6 espacios y se carga una única bala en una de las cámaras al azar. El tambor del revólver se gira para que ningún jugador sepa dónde está la bala. En cada turno, el jugador decide si **se dispara o si dispara al contrincante**.

**Dispararse:** El jugador simula apretar el gatillo del revólver en su turno. Si la cámara está vacía, el jugador sobrevive y puede volver a elegir si dispararse o disparar al contrincante. Si la bala está en la cámara activa, el jugador "pierde" el duelo, la partida y el contrincante se queda con sus casillas.

**Disparar al contrincante:** El jugador decide disparar al oponente, si la bala está en la cámara activa el oponente "pierde" el duelo y el retador se queda con sus casillas, si la cámara está vacía el contrincante sobrevive y se le da el arma para que ahora, este decida si se dispara o dispara al retador.

Esta dinámica genera un juego de alta tensión y equilibrio entre riesgo, estrategia, y lectura del oponente, manteniéndolo interesante para un enfrentamiento 1 vs 1. Inspiración:



## Decisión

Tras una votación democrática, elegimos la “Ruleta Rusa” porque nos pareció la opción más original y emocionante, perfecta para un enfrentamiento 1 vs 1.

## Comenzando a trabajar

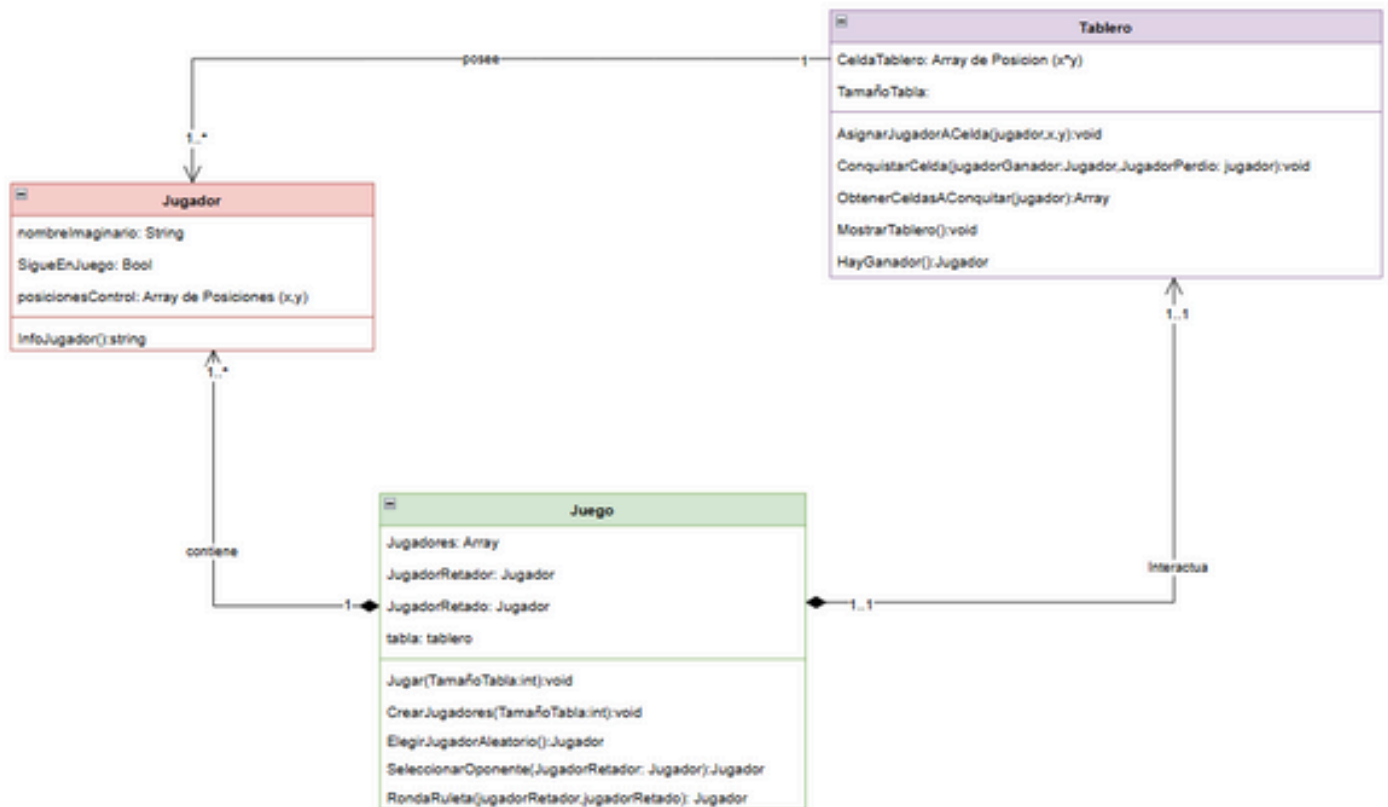
Con las ideas y la estructura general de “La conquista” definidas, nos dedicamos a sentar las bases del juego, comenzando con la organización y planificación detallada de sus elementos principales. Esta etapa fue crucial para alinear la comprensión de todos los miembros y garantizar que trabajamos de manera coordinada.

## Definición de Clases

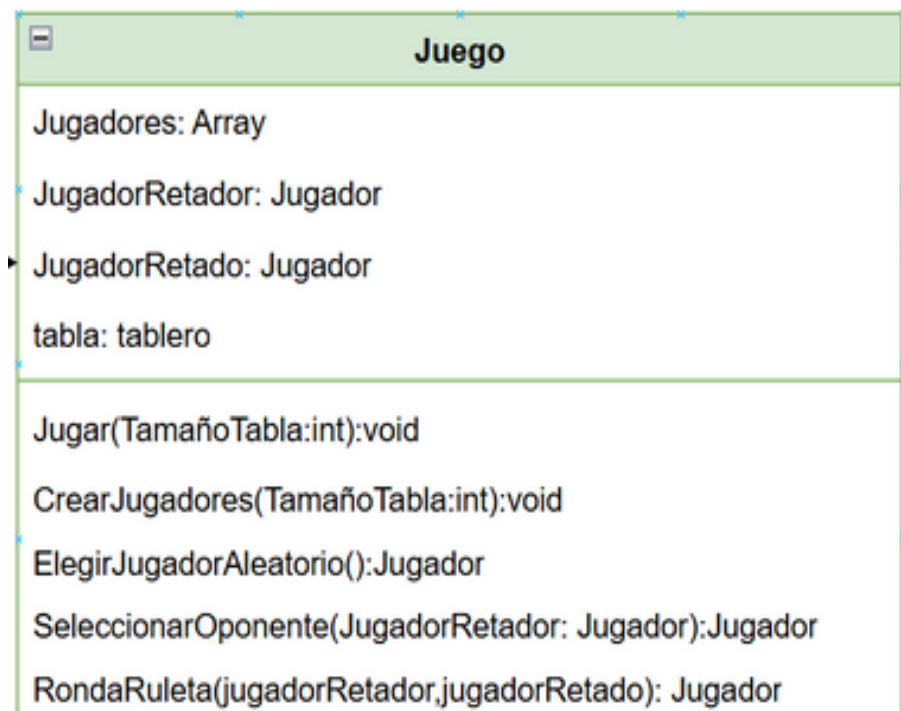
Organizamos una sesión de trabajo en equipo para discutir y decidir qué clases serían esenciales en el desarrollo del juego. Cada clase representa un elemento importante, como los jugadores, el tablero, atributos (características) y métodos (acciones que puede realizar). También dedicamos tiempo a explicar claramente la función de cada componente, asegurándonos de que todos tuviéramos una comprensión común, lo cual quedará plasmado a continuación:

## Diagrama UML

# CONQUITA CON Ruleta RUSA



## Clase Juego



## Atributos:

**Jugadores: Array de Jugador** → se encargará de guardar los datos de cada jugador en el juego, para recurrir a ellos cuando los necesite

**JugadorRetador: Jugador** → guardará los datos del jugador que es seleccionado aleatoriamente.

**JugadorRetado: Jugador** → guardará los datos del jugador al que se le quiere conquistar la casilla.

**Tabla: tablero** → Contiene la instancia de clase Tablero que es el escenario donde sucede el juego y se harán las modificaciones de este.

## Métodos:

**Jugar (TamañoTabla:int)** → Método Utilizado para iniciar el juego y en él se contendrá toda la lógica de control de este.

Recibe como parámetro “TamañoTabla” de tipo entero que determina la longitud de la tabla NxN (cuadrada)

**CrearJugadores(TamañoTabla:int)** →

Es el encargado de rellenar el atributo “Jugadores”, de la clase juego, además de crear la instancia de tablero y colocar a los jugadores creados en la tabla.

Recibe como argumento el tamaño del tablero que se usará para determinar cuántos jugadores podrán participar en el juego

**ElegirJugadorAleatorio(): JugadorRetador** → se encarga de seleccionar aleatoriamente un jugador que será el que puede elegir con quien se enfrentará en el tablero según las posibilidades que tenga

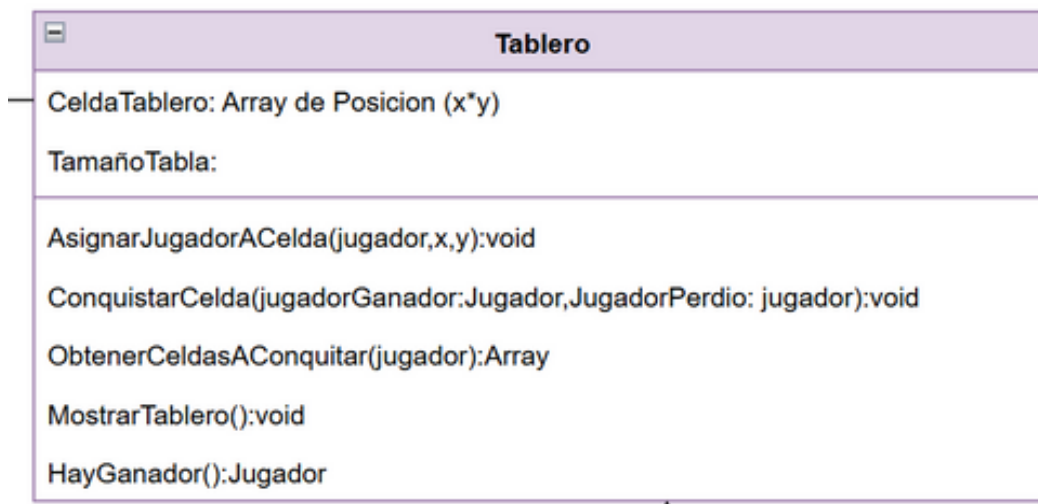
Toma el Array de Jugadores

retorna Aleatoriamente Uno de ellos (Tienen Prioridad los que tengan menos celdas conquistadas)

**TurnoDelJugador(JugadorRetador): JugadorRetado** → toma al jugador determinado retador para que escoja entre los otros jugadores a cual retar, retorna el jugador que fue elegido para ser retado.

**RondaRuleta(JugadorRetador, JugadorRetado): Jugador** → toma al JugadorRetador y al JugadorRetado, en el método se realiza el enfrentamiento entre ambos y retorna el jugador que salió victorioso.

## Clase Tablero



## Atributos:

**CeldaTablero: ArrayDePosiciones(x\*y)** → Simboliza las celdas del tablero donde en cada celda estará un jugador determinado,

**TamañoTabla** → Especifica el tamaño del tablero **MxM**

## Métodos:

**AsignarJugadorACelda(Jugador,x,y)** → Se posiciona el jugador en la fila y columna correspondiente, en caso de que exista un jugador ahí mostrará un mensaje de que “jugador 1 expulsó a jugador 2 “.

recibirá instancia de la clase jugador y las coordenadas donde se asignará a éste donde “X” es la fila e “Y” la columna.

**ObtenerCeldasAConquistar(Jugador):Array** → dará las opciones que tiene el Jugador retador para conquistar, estas serán las casillas que estén contiguas a la casilla del jugador retador.

Recibe como parámetro el jugador retador

devolverá un Array de Arrays en el cual tendrá el siguiente formato:

# (# (jugador, fila,columna) #(jugador,fila,columna) )

esto para poder presentar al jugadorRetador contra quienes pueden enfrentarse y si es el caso de que un mismo jugador este en varias casillas elegir cuál es la casilla que le conviene más.

**MostrarTablero()** → Se llamará a este método cada vez que se quiera visualizar el estado del el tablero actual. Mostrando por pantalla.

**HayGanador():Jugador** → Verificará que haya un solo jugador en el tablero, esto significa que el jugador que hay ha ganado. Retornara el jugador ganador si es el caso.

## Clase Jugador



### Atributos:

(id: Posibilidad de agregar un campo id como atributo de clase.)

**nombreImaginario:String** → Guardo el Nombre del Jugador

**SigueEnJuego: Bool** → Especifica de manera más fácil.

**posicionesControl:** No es necesario creo.

### Métodos:

**InfoJugador()** → Mostrar la información del jugador En caso de que se desee de manera más amigable



### Reunión 3: 31/10/2024

La tercera reunión fue dedicada a revisar y pulir los detalles finales, utilizando el documento compartido en Word, afinamos el contenido del informe, asegurando que cada sección cumpliera con los requisitos de la consigna.

### Planificación para el Desarrollo de la Segunda Entrega

Para llevar a cabo la segunda entrega de nuestro proyecto, decidimos estructurar el trabajo dividiendo las tareas en equipos de dos. Esta estrategia nos permitirá optimizar el tiempo y aprovechar mejor las habilidades de cada miembro del equipo.

- **Equipo 1:** Este grupo se encargará de desarrollar las clases **Tablero** y **Jugador**. Su responsabilidad incluirá la programación de todos los atributos y métodos necesarios para que estas clases sean funcionales y cumplan con los requisitos del juego.
- **Equipo 2:** El segundo grupo se centrará en la clase **Juego**, donde se programarán todos los elementos que gestionan la dinámica del juego en sí. Esto incluirá la implementación de las reglas y la interacción entre las clases previamente mencionadas.

Una vez que ambos equipos hayan completado sus respectivas tareas, se realizará una integración de las diferentes partes del proyecto. Todos los miembros del equipo participarán en la verificación del funcionamiento general, asegurándonos de que todas las clases interactúen correctamente y el juego funcione sin problemas.

De este modo, también fomentamos uno de los principios fundamentales de la Programación Orientada a Objetos (POO): la partición del código. Este enfoque nos permite dividir el programa en componentes más pequeños y especializados, asegurando que cada parte cumpla con funciones específicas y se integre sin problemas en el conjunto del proyecto.

### Expectativas para el final

Actualmente nos encontramos con un objetivo claro sobre cómo queremos que sea el resultado final del juego y sentimos que vamos en la dirección correcta para alcanzarlo. Todavía encontramos algunas dificultades, especialmente en la comprensión y ejecución del lenguaje solicitado, pero confiamos en superar esos desafíos para lograr el resultado final que deseamos.