**Taller GIT: Proyecto Integrador I**

**ENUNCIADO**

Debe desarrollar un juego de ahorcado en un entorno multicliente TCP. El juego se diseñará para que dos jugadores se emparejen y jueguen en turnos alternos. Cada partida constará de varios ciclos donde los roles de los jugadores cambiarán.

**Funcionalidades Principales**

1. Emparejamiento de Jugadores: El sistema debe ser capaz de emparejar automáticamente dos jugadores en una partida. Cada par de jugadores será asignado a una instancia de juego separada.

2. Roles Alternantes: En cada partida, los jugadores asumirán dos roles: el "Creador de Palabra" y el "Adivinador de Letras". En el primer ciclo, uno de los jugadores será el "Creador de Palabra" y el otro el "Adivinador de Letras". En el siguiente ciclo, los roles se intercambiarán.

3. Creador de Palabra: El jugador designado como "Creador de Palabra" debe ingresar una palabra que el otro jugador intentará adivinar. Esta palabra debe ser validada para asegurarse de que sea una palabra válida y no contenga caracteres especiales.

4. Adivinador de Letras: El jugador designado como "Adivinador de Letras" intentará adivinar la palabra ingresando letras una por una. Debe haber una retroalimentación sobre si la letra adivinada está presente en la palabra y en qué posición(es).

5. Representación Visual del Ahorcado: A medida que el "Adivinador de Letras" ingrese letras incorrectas, se mostrará una representación visual del ahorcado en etapas progresivas. Esto servirá como indicador del progreso del juego.

6. Gestión de Ciclos: El juego debe permitir múltiples ciclos de juego dentro de una partida. Después de cada ciclo, los roles de los jugadores cambiarán, y se continuará jugando hasta que se alcance un número predefinido de ciclos.

7. Finalización del Juego: El juego finaliza cuando se completan todos los ciclos preestablecidos. Se mostrará un mensaje indicando el ganador en función del número de adivinaciones correctas en cada ciclo.

Este enunciado describe las funcionalidades clave del juego de ahorcado multicliente, pero está formulado de manera funcional y técnica. Usted y su equipo deben transformar estas funcionalidades en historias de usuario que representen las necesidades de los usuarios y los casos de uso del juego.

**PRODUCT BACKLOG**

1. Escriba las historias de usuario en un proyecto de **Github \***.
2. En la descripción de las historias deben tener los criterios de aceptación. Es decir, qué resultado esperado debe tener la historia para determinarse como completada.
3. Creen su primer sprint.
4. Estimar complejidad y priorizar tareas.
5. Empezar desarrollo con estrategia de branching.

## ESTATEGIA DE BRANCHING

Es necesario crear un repositorio en github con toda la historia del taller

1. Crear branches con el nombre de su historia de usuario ejemplo: **feat:HU1-alfa-beta-gamma**. Deben crear un archivo con la información de readme.md y modificarlo con la información de la historia de usuario que desarrollaron.
2. Deben mezclar las ramas con la rama develop por medio de pull request (resolver conflictos).