

MINI PROYECTO GEEK OUT MASTERS

Nombres: Alan Valderrama – 2042836

Jeison Perea – 2077250

HISTORIAS DE USUARIO

Historia Iniciar Ronda	Criterios de Aceptación
Como jugador deseo iniciar la ronda de juego para ver las caras de los 7 dados activos	<ul style="list-style-type: none">• El área de juego está compuesta de 4 sectores:<ol style="list-style-type: none">1. Dados activos: inicia con 7 dados cuyas caras visibles se seleccionan aleatoriamente.2. Dados Inactivos: inicia con 3 dados, no tienen importancia cual es la cara visible.3. Dados utilizados: inicia vacía4. Marcador de Puntaje: solo muestra los números 1, 3, 6, 10, 15, 21, 28, 36, 45 y 55 (potenciales puntos a ganar en la ronda)• El usuario debe poder acceder a la ayuda explicativa de las acciones, si lo requiere.• El usuario debe poder ver en qué ronda del juego se encuentra.• El usuario debe poder ver los puntos que lleva acumulados si previamente a jugado una ronda.• El jugador debe poder en cualquier momento salir del juego si lo desea.

Historia Ejecutar Acciones	Criterios de Aceptación
<p>Como jugador deseo seleccionar dados de la zona de dados activos para utilizarlos y saber si logro sumar puntos en la ronda de juego y ganar el juego.</p>	<ul style="list-style-type: none"> • Los dados de la zona de dados activos se seleccionan de uno por vez. • Un dado seleccionado de la zona de dados activos debe pasar a la zona de dados utilizados y solo debe permitir ejecutar la acción relacionada con la cara visible del dado seleccionado. • Las posibles acciones a realizar según la cara visible del dado son: <ol style="list-style-type: none"> 1. Meeple: permite al jugador poder volver a lanzar uno de los dados activos. 2. Cohete: permite al jugador eliminar un dado pasándolo al área de dados inactivos. 3. Superhéroe: permie al jugador dar vuelta (mostrar la cara opuesta) de un dado de la zona de dados activos. Un Superhéroe no puede voltearse a sí mismo. 4. Corazón: permite al jugador tomar un dado de la zona de dados inactivos y lanzarlo en la zona de dados activos. 5. Dragón: Si es el último de los dados activos, hace que el jugador pierda todos los puntos que lleve hasta el momento. 6. 42: Si el último o últimos dados activos tiene esta cara visible, se suman dichos puntos en el tablero de puntaje, acorde con las reglas indicadas, generando el puntaje de la ronda. • Las caras contrarias del dado se corresponden con el color que comparten: <ol style="list-style-type: none"> 1. Meeple y Cohete son contrarios 2. Dragón y Superhéroes son contrarios 3. Corazón y 42 son contrarios • Una vez el usuario termine de ejecutar las acciones que le permiten sus dados activos, debe calcularse los puntos logrados en la ronda de juego y se debe permitir al jugador iniciar una nueva ronda de juego o indicar el estado del juego (ganar o perder).

	<ul style="list-style-type: none"> Los puntos logrados en la ronda de juego se determinar chequeando qué dados permanecen en el área de dados activos, cuándo ya no se puede ejecutar más acciones. Teniendo en cuenta que: <ol style="list-style-type: none"> Si sólo hay dados con la cara 42, estos se registran en el área de dados puntuados, para saber los puntos logrados en la ronda. Si solo hay dragones el jugador no obtiene ningún punto y pierde incluso sus puntos acumulados. Si no queda ningún dado en la zona de dados activos, o Si el último dado de la zona de dados activos es un Meeple, un Superhéroe o un Cohete, la ronda de juego termina sin ganar puntos y se conservan los puntos que se tengan acumulados. El jugador podrá iniciar una ronda de juego si no ha completado el máximo de 5 rondas y tiene menos de 30 puntos. El jugador ganará si alcanza los 30 puntos en máximo 5 rondas de juego. El Jugador perderá si terminada la 5ta ronda de juego tiene menos de 30 puntos acumulados. Si el jugador gana o pierde podrá reiniciar el juego. El usuario debe poder acceder a la ayuda explicativa de las acciones, si lo requiere. El usuario debe poder ver en qué ronda del juego se encuentra. El usuario debe poder ver los puntos que lleva acumulados si previamente a jugado una ronda. El jugador debe poder en cualquier momento salir del juego si lo desea.
--	---

DISEÑO DE CLASES

MODEL

Dice	
Responsabilidades Devolver el valor de la cara visible del dado.	Colaboraciones GeekOutMasters

GeekOutMasters	
Responsabilidades Determinar la cara de los dados cada vez que el jugador presione el botón "Play". Determinar en que ronda se encuentra el jugador. Ejecutar los diferentes poderes que hay en el juego. Determinar los puntos que ha hecho el jugador. Determinar si el jugador perdió o ganó.	Colaboraciones Escucha

CONTROLLER

<<EventListener>> Listener	
Responsabilidades Determinar cuál dado está presionando el jugador. Actualizar la GUI cuando se ejecute un movimiento para evitar sobrecargas para los diferentes paneles.	Colaboraciones GeekOutMasters

VIEW

<<JFrame>> GUIGridBagLayout	
Responsabilidades Mostrar los diferentes paneles, botones, dados e información que contiene el juego. Mostrar el estado del juego.	Colaboraciones Escucha

BOCETO

Geek Out Master Table

Columns: 0, 1, 2

Rows: 0, 1, 2, 3, 4

How to Play?

Inactive dice

Dice used

Your score

Your dice

Play

0, 1, 2, 3, 4