**DOCUMENTACIÓN CASOS DE USO**

**Alimnova®**

**Descripción**

Este documento tiene como finalidad mostrar la documentación de los casos de uso identificados y propuestos por AlimNova®, también es importante tener en cuenta los documentos:

* Diagrama de casos de uso
* Glosario de actores y responsabilidades

Donde están específicos los actores correspondientes a cada caso de uso y los diagramas correspondientes a cada uno de los casos de uso.

Los casos de uso están ordenados acorde a las prioridades dadas en los puntos de función de cada uno.

El formato empleado para realizar la documentación de cada caso de uso es:

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | 1 | *Nombre:* | 2 |
| *Nivel*  *(prioridad)* | |  | |
| *Garantía Mínima* | |  | |
| *Objetivo en Contexto:* | | 5 | |
| *Actores Participantes* | | 5 | |
| *Pre-Condiciones* | | 7 | |
| *Post-Condiciones* | | *Condición final de éxito: 8* | |
| *Condiciones finales de fallo: 8* | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito 9* | | | |
|  | *Actor* |  | *Sistema* |
|  |  |  |  |

|  |  |
| --- | --- |
| *Variaciones* | 10 |

En donde:

1. **Id Caso de Uso:** Corresponde al identificador único de cada caso de uso.
2. **Nombre:** Describe el objetivo del caso de uso como verbo activo corto.
3. **Nivel:** Relacionado con la medida del objetivo del caso de uso a tratar (alta, baja), acorde a la prioridad definida para cada uno.
4. **Garantía mínima:** Describe las pequeñas promesas que el sistema hace a los stakeholders particularmente cuando el objetivo no puede ser llevado a cabo en su totalidad.
5. **Objetivo en Contexto**: Expresa que es lo que va a realizar determinado caso de uso en el juego (estado, presentación), teniendo en cuenta la garantía mínima.
6. **Actores Participantes:** Representa los participantes involucrados directamente con el caso de uso.
7. **Pre-Condiciones:** Describe las condiciones mínimas que deben ser verdaderas antes de ejecutar el caso de uso.
8. **Post-Condiciones:** Describe las condiciones que deben ser verdaderas luego de la ejecución del caso de uso

**6.1 Condición final de éxito:** Correspondeal cumplimiento del objetivo del caso de uso, luego de su ejecución.

**6.2** **Condiciones finales de fallo:** Correspondea una falla en el sistema, que lleva a nocumplir con el objetivo del caso de uso.

1. **Flujo básico de éxito:** Describe el flujo de información durante la ejecución del caso de uso.

**7.1 Actor:** Aquel que interactúa con el sistema, para satisfacer sus necesidades.

**7.2 Sistema:** Corresponde a las actividades que requiere realizar el servidor para actualizar, cambiar, informar, determinados estados del juego.

1. **Variaciones:** Condiciones sobre las cuales el sistema realiza ciertos comportamientos.

Acorde al número de variaciones en cada caso de uso, dependiendo el número de variaciones se va a enumerar acorde al paso correspondiente que podría variar, seguido por un punto y el número desde 1, correspondiente si es la primera, segunda, etc variación.

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU01 | *Nombre:* | Inicializar partida |
| *Nivel*  *(prioridad)* | | 5 | |
| *Garantía Mínima* | | Generar la conexión entre los jugadores que desean inicializar el juego | |
| *Actores Participantes* | | Sistema | |
| *Objetivo en Contexto:* | | Disponer de los elementos necesarios para que los jugadores puedan comenzar la partida (tablero, fichas, propiedades). | |
| *Pre-Condiciones* | | Jugador debe tener instalado el juego T-monopoly.  Jugador debe haber creado un perfil (crear perfil) | |
| *Post-Condiciones* | | *Condición final de éxito:* A cada jugador se le asigna una ficha de juego, se le muestra tablero y se le asignan 2 propiedades. | |
| *Condiciones final de fallo:*  No se le muestra TDP correctamente, con las propiedades y las fichas.  No se le asigna una ficha al jugador.  No se genera conexión al jugador. | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 1 | Abrir el programa | 2 | Pregunta tipo de acción a realizar |
| 3 | Escoge inicializar partida | 4 | Pregunta nombre |
| 5 | Ingresa el nombre | 7 | Verifica contraseña y nombre |
| 6 | Ingresa contraseña | 8 | Muestra las fichas disponibles |
| 9 | Escoge ficha | 10 | Asigna dos propiedades |
| 10 | Pulsa botón jugar | 11 | Muestra tablero de juego |

|  |  |
| --- | --- |
| *Variaciones* | 7.1: En caso de que el nombre del perfil ingresado por el usuario se encuentre repetido, se le informa al usuario, no se guarda perfil y se le pregunta por otro nombre.  9.1: En caso de que la ficha ya la haya pedido otro jugador, se le informa al usuario y se le pregunta por otra ficha. |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU02 | *Nombre:* | Subastar |
| *Nivel*  *(prioridad)* | | 5 | |
| *Garantía Mínima* | | Informar que se va a realizar la subasta a los jugadores | |
| *Objetivo en Contexto:* | | Vender la propiedad en la cual un jugador haya caído y no desee o no pueda comprarla | |
| *Actores Participantes* | | Banco  Jugador | |
| *Pre-Condiciones* | | Jugador debió caer en propiedad libre  Jugador no puede comprarla por falta de recursos  Jugador no quiere comprarla  Jugadores a participar deben tener el dinero del precio base como mínimo | |
| *Post-Condiciones* | | *Condiciones final de éxito:* Se le vende la propiedad al jugador que haya ofrecido más dinero | |
| *Condiciones finales de fallo:*  Sistema no tiene en cuenta todos los jugadores dispuestos a pagar  La propiedad subastada tiene dueño | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 4 | Acepta entrar a la subasta | 1 | Notifica a los jugadores sobre la subasta |
| 5 | Ofrece cantidad de dinero | 2 | Estipula precio base |
|  |  | 3 | Pregunta quienes quieren entrar a la subasta |
|  |  | 6 | Escoge mayor cantidad de dinero ofrecido |
|  |  | 7 | Asigna propiedad a mayor oferta |

|  |  |
| --- | --- |
| *Variaciones* | 4.1 El jugador no acepta entrar en la subasta.  4.1.1 El sistema verifica existen subastadores.  6.1 Solo hay un jugador en la subasta.  6.1.1 Se le asigna la propiedad a él jugador.  6.2 No existen jugadores en la subasta  6.2.1 El banco toma la propiedad. |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU02 | *Nombre:* | Subastar |
| *Nivel*  *(prioridad)* | | 5 | |
| *Garantía Mínima* | | Informar que se va a realizar la subasta a los jugadores | |
| *Objetivo en Contexto:* | | Vender la propiedad en la cual un jugador haya caído y no desee o no pueda comprarla | |
| *Actores Participantes* | | Banco  Jugador | |
| *Pre-Condiciones* | | Jugador debió caer en propiedad libre  Jugador no puede comprarla por falta de recursos  Jugador no quiere comprarla  Jugadores a participar deben tener el dinero del precio base como mínimo | |
| *Post-Condiciones* | | *Condiciones final de éxito:* Se le vende la propiedad al jugador que haya ofrecido más dinero | |
| *Condiciones finales de fallo:*  Sistema no tiene en cuenta todos los jugadores dispuestos a pagar  La propiedad subastada tiene dueño | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 4 | Acepta entrar a la subasta | 1 | Notifica a los jugadores sobre la subasta |
| 5 | Ofrece cantidad de dinero | 2 | Estipula precio base |
|  |  | 3 | Pregunta quienes quieren entrar a la subasta |
|  |  | 6 | Escoge mayor cantidad de dinero ofrecido |
|  |  | 7 | Asigna propiedad a mayor oferta |

|  |  |
| --- | --- |
| *Variaciones* | 4.1 El jugador no acepta entrar en la subasta.  6.1 Solo hay un jugador en la subasta.  6.1.1 Se le asigna la propiedad a él jugador.  6.2 No existen jugadores en la subasta  6.2.1 El banco toma la propiedad. |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU03 | *Nombre:* | Comprar tragos |
| *Nivel*  *(prioridad)* | | 5 | |
| *Garantía Mínima* | | Habilitar al jugador la opción de comprar tragos | |
| *Objetivo en Contexto:* | | Asignar al jugador el trago que el desee en su mesa sea cerveza ó una botella de Sello azul. | |
| *Actores Participantes* | | Jugador  Banco | |
| *Pre-Condiciones* | | Jugador debe tener dinero para comprar la cantidad de trago que pide.  Jugador debe ser mayor de edad, acorde a las condiciones del juego. ( ver glosario)  Debe ser el turno del jugador  Debe pedir hasta de 4 cervezas para una sola propiedad y una botella para propiedades diferentes. | |
| *Post-Condiciones* | | *Condición final de éxito:* Asignación de el número de cervezas ó botella de Sello azul que pidió | |
| *Condiciones finales de fallo:* No se le asigna el número de cervezas ó botella de Sello azul que pidió el jugador | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
|  |  | 1 | Habilita opción de compra de trago al jugador |
| 2 | Jugador realiza pedido de trago | 3 | Verifica cantidad de tragos |
|  |  | 4 | Notifica jugador acerca del valor del pedido |
| 6 | Aprueba cantidad a pagar | 5 | Espera aprobación del usuario |
|  |  | 7 | Descuenta dinero |
|  |  | 8 | Asigna trago |

|  |  |
| --- | --- |
| *Variaciones* | 3. 1: Cantidad de tragos inválida  3.1.1: Notifica que no puede realizar esa cantidad en su pedido  6.1: Jugador no aprueba pago de esa cantidad de dinero |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU04 | *Nombre:* | Hipotecar propiedad |
| *Nivel*  *(prioridad)* | | 5 | |
| *Garantía Mínima* | | Habilitar al jugador la opción de hipotecar cierta propiedad | |
| *Objetivo en Contexto:* | | El jugador pone en hipoteca una propiedad, es decir, que el banco le presta el dinero acorde al valor de la hipoteca de la propiedad | |
| *Actores Participantes* | | Jugador  Banco | |
| *Pre-Condiciones* | | Jugador debe tener propiedades  Debe ser el turno del jugador para realizar esta operación  Propiedades a hipotecar no deben tener tragos | |
| *Post-Condiciones* | | *Condición final de éxito:* Jugador hipoteca su propiedad y recibe el dinero correspondiente de esta, por el banco | |
| *Condiciones finales de fallo:*  Jugador no puede hipotecar  Jugador no recibe dinero del ba*nco* | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 1 | Jugador informa que quiere hipotecar determinada propiedad | 2 | Evalúa que si sea su propiedad |
|  |  | 3 | Identifica precio de la hipoteca de la propiedad |
|  |  | 4 | Asigna el valor de la hipoteca, como préstamo |
|  |  | 5 | Solicita confirmación de la hipoteca |
| 6 | Confirma deseo de hipotecar | 7 | Pone la propiedad como hipotecada |

|  |  |
| --- | --- |
| *Variaciones* | 6.1: Jugador no confirma que quiere hipotecar  6.1.1: Continúa secuencia de juego |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU05 | *Nombre:* | Negociar |
| *Nivel*  *(prioridad)* | | 5 | |
| *Garantía Mínima* | | Habilitar al jugador la opción de negociar con otro jugador | |
| *Objetivo en Contexto:* | | Negociar entre los jugadores las propiedades que cada quién desea a cambio de propiedades, dinero, tragos, etc, acorde los jugadores acuerden | |
| *Actores Participantes* | | Jugador1  Jugador2 | |
| *Pre-Condiciones* | | Jugadores deben tener propiedades, dinero para negociar, quién inicia negociación debe tener el turno para hacerlo | |
| *Post-Condiciones* | | *Condición final de éxito:* Se realiza el intercambio entre las propiedades que requiere determinado jugador | |
| *Condiciones finales de fallo:*  No se intercambien correctamente el dinero  No se intercambien correctamente las propiedades | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 1 | Jugador1 manifiesta que quiere negociar con jugador2 | 2 | Muestra propiedades de ambos jugadores |
| 3 | Jugador1 realiza su oferta al jugador2 |  |  |
| 4 | Jugador2 analiza oferta y acepta | 5 | Intercambia propiedades |

|  |  |
| --- | --- |
| *Variaciones* | 4.1: Jugador2 no acepta oferta de Jugador1  4.1.1: Realiza nueva oferta |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU06 | *Nombre:* | Actualizar tablero |
| *Nivel*  *(prioridad)* | | 5 | |
| *Garantía Mínima* | | Mostrar estado actual de las fichas en el tablero | |
| *Objetivo en Contexto:* | | Poner las fichas acorde a la última posición, las propiedades disponibles y las que tienen dueño todo esto según la última jugada | |
| *Actores Participantes* | | Sistema | |
| *Pre-Condiciones* | | Debe estar una jugada activa  Haber realizado una jugada previa a la actualización | |
| *Post-Condiciones* | | *Condición final de éxito:* Usuarios pueden ver la situación actual del juego en la pantalla | |
| *Condiciones finales de fallo:*  No se pueden ver las últimas jugadas  No se puede ver la ubicación actual de las fichas  No se puede ver estado actual del juego | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 1 | Jugador termina turno | 2 | Actualiza la disposición de las fichas, las propiedades compradas, etc |
|  |  | 3 | Muestra tablero actualizado a todos los jugadores |

|  |  |
| --- | --- |
| *Variaciones* | 3.1: No muestra tablero actualizado  3.1.1: Actualiza en siguiente turno estado del tablero |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU07 | *Nombre:* | Verificar jugador ganador |
| *Nivel*  *(prioridad)* | | 5 | |
| *Garantía Mínima* | | Realizar el cálculo del valor de las posesiones de cada jugador | |
| *Objetivo en Contexto:* | | Comparar la suma del valor de las posesiones de los jugadores para poder determinar el ganador del juego | |
| *Actores Participantes* | | Sistema | |
| *Pre-Condiciones* | | Declaración de bancarrota por parte de un jugador | |
| *Post-Condiciones* | | *Condición final de éxito:* Mostrar el jugador ganador de la partida | |
| *Condiciones finales de fallo:*  Realizar proceso sin que un jugador se haya declarado en banca rota  Suma de posesiones de cada jugador errónea  Información errónea acerca del ganador | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
|  |  | 1 | Sumar valor de todas las posesiones de cada jugador |
|  |  | 2 | Buscar el jugador cuya suma del valor de las posesiones sea mayor |
|  |  | 3 | Informar a todos los jugadores que hay un ganador mostrando el perfil del mismo |

|  |  |
| --- | --- |
| *Variaciones* | 2.1: Hay dos jugadores con igual cantidad de posesiones, siendo estas las mayores. (empate)  2.1.1: Jugadores lanzan los dados y gana el de mayo número |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU08 | *Nombre:* | Deshipotecar |
| *Nivel*  *(prioridad)* | | 5 | |
| *Garantía Mínima* | | Permitir al jugador la opción de deshipotecar la propiedad que desee | |
| *Objetivo en Contexto:* | | Lograr que el banco le devuelva una propiedad que se encuentre hipotecada al propietario luego de que este haya pagado la hipoteca | |
| *Actores Participantes* | | Banco  Jugador | |
| *Pre-Condiciones* | | Debe existir una propiedad hipotecada del jugador  Jugador debe ser el turno para poder deshipotecar  Jugador deshipoteca solamente la propiedad que le pertenece  Jugador debe tener cantidad de dinero suficiente para cancelar la hipoteca | |
| *Post-Condiciones* | | *Condición final de éxito:* Luego de cancelar el valor de la hipoteca, la propiedad de ese jugador queda libre | |
| *Condiciones finales de fallo:*  Jugador no puede recuperar su propiedad hipotecada  Jugador cancele y no reciba propiedad  Reciba propiedad sin cancelación de hipoteca | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 1 | Enviar mensaje de desear pagar la hipoteca | 2 | Muestra propiedades hipotecadas por el jugador |
| 3 | Selecciona propiedad que desea deshipotecar | 4 | Muestra el precio de la hipoteca |
|  |  | 5 | Pregunta si está dispuesto a pagar esa cantidad |
| 6 | Acepta cancelación de esa cantidad por la hipoteca | 7 | Debita cantidad |
|  |  | 8 | Deshipoteca propiedad |

|  |  |
| --- | --- |
| *Variaciones* | 6.1: Jugador no acepta cancelar la hipoteca  6.1.1: Propiedad continúa hipotecada por el banco |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU09 | *Nombre:* | Guardar perfil de usuario |
| *Nivel*  *(prioridad)* | | 5 | |
| *Garantía Mínima* | | Pide información del perfil del jugador | |
| *Objetivo en Contexto:* | | Hacer que el sistema guarde los perfiles creados por los jugadores | |
| *Actores Participantes* | | Sistema | |
| *Pre-Condiciones* | | Jugador debe tener un perfil  Partida debe estar iniciada | |
| *Post-Condiciones* | | *Condición final de éxito:* Guardar la información correspondiente al perfil de un jugador nuevo | |
| *Condiciones finales de fallo:*  El perfil de usuario ya se encuentra creado  No se puede guardar perfil de usuario | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 2 | Ingresa el nombre | 1 | Pide nombre de usuario |
| 4 | Ingresa contraseña | 3 | Pide contraseña |
|  |  | 5 | Almacena información del perfil de usuario ingresado |

|  |  |
| --- | --- |
| *Variaciones* | 4.1:No se puede crear perfil  4.1.1: Se le informa al jugador el problema con el almacenamiento del perfil |
|  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU10 | *Nombre:* | Guardar historial del juego |
| *Nivel*  *(prioridad)* | | 5 | |
| *Garantía Mínima* | | Permitirle al usuario la opción de guardar el historial de su juego | |
| *Objetivo en Contexto:* | | Guardar la información de las jugadas realizadas por cada persona en el juego durante cada turno; hacia donde se dirigió con cada lanzamiento, si compró propiedad o no, si pagó alquiler, entre otras situaciones que se pueden generar durante el juego | |
| *Actores Participantes* | | Sistema | |
| *Pre-Condiciones* | | Juego debe estar activo  Jugador debe estar en turno  Jugador debe realizar determinada acción para que esta sea registrada | |
| *Post-Condiciones* | | *Condición final de éxito:* Se genera una un historial de cada uno de los movimiento realizados por el jugador, durante el juego | |
| *Condiciones final de fallo:*  No se genera el historial  Problemas con el archivo para guardar el historial  No se registren todas las jugadas realizadas durante un turno | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 1 | Jugador informa que quiere mantener almacenado su historial durante el turno terminado | 2 | Almacena jugada realizada por el jugador (compra propiedad, hipoteca, pago de cover, etc) |
|  |  | 3 | Muestra resultado de cada jugada realizada por el jugador que terminó turno |

|  |  |
| --- | --- |
| *Variaciones* | 1.1:Jugador no quiere almacenar su historial durante cierto turno  1.1.1: No se muestran sus actividades durante ese turno |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU11 | *Nombre:* | Comprar propiedad |
| *Nivel*  *(Prioridad)* | | 5 | |
| *Garantía mínima* | | Ofrecerle una propiedad al jugador | |
| *Objetivo en Contexto:* | | Asignar al jugador una propiedad por la cual haya pagado | |
| *Actores Participantes* | | Jugador  Banco | |
| *Pre-Condiciones* | | Jugador debe tener dinero suficiente para cancelar la cantidad pedida para comprar la propiedad  La casilla en la que cae el jugador debe no tener dueño, es decir debe estar disponible para la venta  Debe ser el turno del jugador | |
| *Post-Condiciones* | | *Condición final de éxito:* Asignación de la propiedad pedida al jugador | |
| *Condiciones finales de fallo:* No se le da la propiedad al jugador  Se le cobra más dinero del que debe pagar | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 3 | Jugador confirma compra de la propiedad | 1 | Muestra información correspondiente a la propiedad |
|  |  | 2 | Pregunta si desea comprar propiedad |
|  |  | 4 | Debita el valor correspondiente a esa propiedad |
|  |  | 5 | Asigna propiedad al jugador |

|  |  |
| --- | --- |
| *Variaciones* | 3.1:Jugador no quiere comprar la propiedad  3.1.1: Se subasta la propiedad correspondiente |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU12 | *Nombre:* | Disminuir tiempo de jugada |
| *Nivel*  *(prioridad)* | | 5 | |
| *Garantía mínima* | | Mostrar contador correspondiente al tiempo restante de jugada que queda | |
| *Objetivo en Contexto:* | | Disminuir el tiempo dado a que el jugador termine de realizar determina actividad en el juego (lanzar dados, negociación, etc) | |
| *Actores Participantes* | | Sistema | |
| *Pre-Condiciones* | | Juego debe estar activo  Debe ser el turno de algún jugador | |
| *Post-Condiciones* | | *Condición final de éxito:* Se disminuye el tiempo correspondiente a una jugada | |
| *Condiciones finales de fallo:*  Sistema no disminuye tiempo de jugada | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
|  |  | 1 | Muestra pantalla con el tiempo que tiene de jugada |
|  |  | 2 | Actualiza tiempo de jugada |
|  |  | 3 | Evalúa el tiempo transcurrido |

|  |  |
| --- | --- |
| *Variaciones* | 3.1:Tiempo es cero  3.1.1: Notifica que el tiempo para la jugada se ha terminado |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU13 | *Nombre* | Cobrar cover |
| *Nivel*  *(prioridad)* | | 5 | |
| *Garantía mínima* | | Informar al jugar acerca del valor del cover | |
| *Objetivo en Contexto:* | | Cobrar a un jugador el valor correspondiente al cover por caer en una propiedad que es de otro jugador | |
| *Actores Participantes* | | Jugador  Sistema | |
| *Pre-Condiciones* | | El juego debe esta inicializado.  Los dos jugadores deben estar en el juego.  La propiedad debe ser de un jugador específico, diferente al que cayó ahí en determinado punto  La propiedad no puede estar hipotecada | |
| *Post-Condiciones* | | *Condición final de éxito:* Jugador dueño de cierta propiedad recibe el dinero correspondiente al cover de su propiedad | |
| *Condiciones final de fallo:* No se realiza cobro sobre determinada propiedad  Se cobra más dinero que el correspondiente al cover  No se asigna dinero de cover al dueño de la propiedad | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
|  |  | 1 | Muestra cantidad correspondiente al cover de la propiedad |
|  |  | 2 | Verifica fondos del jugador que debe pagar |
|  |  | 3 | Notifica cobro de esa cantidad al jugador |
| 4 | Acepta dicho cobro | 5 | Sistema verifica quién es el dueño de esa propiedad |
|  |  | 6 | Transfiere cantidad del cover al dueño de la propiedad |

|  |  |
| --- | --- |
| *Variaciones* | 2.1:Jugador no cuenta con fondos suficientes para cancelar  2.1.1: Hipoteca propiedades si tiene  2.2:No tiene propiedades para hipotecar  2.2.1: Declara bancarota |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU14 | *Nombre:* | Asignar dinero inicial |
| *Nivel*  *(prioridad)* | | 4 | |
| *Garantía mínima* | | Asignar una cantidad de dinero a cada jugador | |
| *Objetivo en Contexto:* | | Darle a cada jugador determinada cantidad de dinero, para iniciar el juego | |
| *Actores Participantes* | | Banco | |
| *Pre-Condiciones* | | Jugadores deben haber iniciado partida  Jugador deben tener perfil  Jugadores ingresaron al juego con su perfil | |
| *Post-Condiciones* | | *Condición final de éxito:* A todos los jugadores se les asignó la misma cantidad de dinero | |
| *Condiciones final de fallo:*  No se asigna dinero a algún jugador  No se asignan dinero a ningún jugador  No se asigna cantidad que debe darse a cada jugador | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 1 | Jugador se registra en la partida | 2 | Verifica que este registrado |
|  |  | 3 | Asigna valor determinado de dinero para iniciar partida |

|  |  |
| --- | --- |
| *Variaciones* |  |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU15 | *Nombre:* | Escoger tarjeta de arca comunal |
| *Niveles*  *(prioridad)* | | 4 | |
| *Garantía mínima* | | Mostrar el mensaje de la tarjeta de arca comunal correspondiente a la casilla en la que quedó el jugador | |
| *Objetivo en Contexto:* | | Mostrar al jugador la tarjeta de arca comunal correspondiente | |
| *Actores Participantes* | | Jugador  Sistema | |
| *Pre-Condiciones* | | Juego debe estar activo  El juego debe estar inicializado.  Debe ser el turno del jugador | |
| *Post-Condiciones* | | *Condición final de éxito:*  *Se le asigna una tarea obligatoria al jugador al caer sobre una casilla*  de arca comunal. | |
| *Condiciones final de fallo:*  *No es posible asignar una tarea obligatoria al jugador al caer sobre una casilla*  de arca comunal. | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 2 | Jugador recibe una tarea obligatoria de “arca comunal”. | 1 | Sistema verifica si la casilla en la que está es la casilla de arca comunal |
|  |  | 3 | El sistema ofrece al jugador una tarea de arca comunal aleatoriamente |

|  |  |
| --- | --- |
| *Variaciones* |  |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU16 | *Nombre:* | Escoger tarjeta de casualidad |
| *Niveles*  *(prioridad)* | | 4 | |
| *Garantía mínima* | | Mostrar el mensaje de la tarjeta de casualidad correspondiente a la casilla en la que quedó el jugador | |
| *Objetivo en Contexto:* | | Mostrar al jugador la tarjeta de causalidad correspondiente | |
| *Actores Participantes* | | Jugador  Sistema | |
| *Pre-Condiciones* | | Juego debe estar activo  El juego debe estar inicializado.  Debe ser el turno del jugador | |
| *Post-Condiciones* | | *Condición final de éxito:*  *Se le asigna una tarea obligatoria al jugador al caer sobre una casilla*  de casualidad | |
| *Condiciones final de fallo:*  *No es posible asignar una tarea obligatoria al jugador al caer sobre una casilla*  de casualidad | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 2 | Jugador recibe una tarea obligatoria de “casualidad”. | 1 | Sistema verifica si la casilla en la que está es la casilla casualidad |
|  |  | 3 | El sistema ofrece al jugador una tarea de casualidad aleatoriamente |

|  |  |
| --- | --- |
| *Variaciones* |  |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU17 | *Nombre:* | Cobrar parqueaderos |
| *Nivel*  *(prioridad)* | | 4 | |
| *Garantía mínima* | | Calcular cobro total, acorde al número de parqueaderos que tenga | |
| *Objetivo en Contexto:* | | Cobro especial acorde al número de parqueaderos que tenga el jugador | |
| *Actores Participantes* | | Jugador  Sistema | |
| *Pre-Condiciones* | | El jugador debe estar es su correspondiente turno.  Jugador debe tener parqueaderos  Jugador debe tener el turno de lanzamiento  Jugador que cae no es dueño de ese parqueadero | |
| *Post-Condiciones* | | *Condición final de éxito:*  Se realiza el cobro del parqueadero | |
| *Condiciones finales de fallo:* No se cancela el valor correspondiente al cobro del parqueadero  Se cobra mal el parqueadero | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
|  |  | 1 | El sistema verifica si esta sobre una casilla de parqueadero |
|  |  | 2 | Identifica dueño del parqueadero y número de parqueaderos que tiene |
|  |  | 3 | Calcula valor correspondiente al pago del parqueadero |
|  |  | 4 | Muestra total a pagar por el jugador |
|  |  | 5 | Transfiere valor del parqueadero al dueño del mismo |

|  |  |
| --- | --- |
| *Variaciones* |  |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU018 | *Nombre:* | Cobrar servicios públicos |
| *Nivel*  *(prioridad)* | | 4 | |
| *Garantía mínima* | | Calcular el cobro total a pagar por servicios públicos | |
| *Objetivo en Contexto:* | | Cobrar al jugador que caiga en la casilla de servicios público acorde al número de propiedades que tenga | |
| *Actores Participantes* | | Sistema  Jugador | |
| *Pre-Condiciones* | | Juego debe estar activo  Casilla de servicio público debe tener dueño  Jugador que cae en casilla de servicio público no es el dueño de esta | |
| *Post-Condiciones* | | *Condición final de éxito:* Realizar el cobro apropiado para los servicios públicos | |
| *Condiciones final de fallo:* Jugador no tienes fondos suficientes para cancelar los servicios públicos  Se le cobra más al jugador que lo que debe  No se le cobra al jugador | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 5 | Acepta pago de servicios públicos | 1 | Identifica número de propiedades del jugador que debe pagar en la casilla de servicios públicos |
|  |  | 2 | Evalúa fondos del jugador |
|  |  | 3 | Notifica al jugador el valor a pagar |
|  |  | 4 | Esperar confirmación de pago por parte del jugador |
|  |  | 6 | Recibe confirmación de pago |
|  |  | 7 | Transfiere valor de los servicios públicos al dueño de esta propiedad |

|  |  |
| --- | --- |
| *Variaciones* | 1.1:Jugador no tiene propiedades  1.1.1: Jugador debe pagar precio base  1.2:Jugador tiene propiedades hipotecadas  1.2.1: No se cobra por esa propiedad  2.1:Jugador no tiene fondos para cancelar los servicios públicos  2.1.1: Preguntar si está en bancarrota |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU19 | *Nombre:* | Vender trago al banco |
| *Nivel*  *(prioridad)* | | 4 | |
| *Garantía mínima* | | Habilitar la opción de venderle trago al banco | |
| *Objetivo en Contexto:* | | El jugador le vende al banco el trago del que dispone y que quiere ofrecer | |
| *Actores Participantes* | | Jugador  Banco | |
| *Pre-Condiciones* | | Jugador debe tener trago para ofrecer  Debe ser el turno del jugador para realizar este ofrecimiento | |
| *Post-Condiciones* | | *Condición final de éxito:* Jugador vende al banco el trago y recibe el dinero correspondiente a esa transacción | |
| *Condiciones final de fallo:* No se realiza la venta del trago al banco  No se cancela dinero al jugador  No recibe dinero el jugador que ha realizado la venta | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 1 | Jugador informa al banco que quiere venderle determinado trago | 2 | Banco muestra el valor que está dispuesto a pagar por el trago ofrecido |
| 3 | Jugador acepta propuesta | 3 | Paga por el trago ofrecido |

|  |  |
| --- | --- |
| *Variaciones* | 3.1: Jugador no acepta propuesta  3.1.1: No se hace ningún negocio |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU20 | *Nombre:* | Delegar turno |
| *Nivel*  *(prioridad)* | | 3 | |
| *Garantía mínima* | | Informar al jugador cuando le corresponda su turno | |
| *Objetivo en Contexto:* | | Asignar el turno del usuario al momento de jugar | |
| *Actores Participantes* | | Sistema | |
| *Pre-Condiciones* | | Jugador debe tener un perfil de usuario y una ficha asignada  Juego debe estar activo  Jugador no debe estar en bancarrota | |
| *Post-Condiciones* | | *Condición final de éxito:* Jugador conoce el turno que va a tener a lo largo del juego | |
| *Condiciones final de fallo:* Jugador notiene turno asignado | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
|  |  | 1 | Busca turno disponible |
|  |  | 2 | Asigna turno al jugador |

|  |  |
| --- | --- |
| *Variaciones* | 1.1: No hay turno disponible  1.1.1: Informar al jugador que el número de jugadores para la partida ya está completo |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU21 | *Nombre:* | Mover ficha |
| *Nivel*  *(prioridad)* | | 3 | |
| *Garantía mínima* | | Cambiar de ubicación la ficha del jugador | |
| *Objetivo en Contexto:* | | Desplazar el jugador de su posición a otra casilla acorde al número que saque al lanzar los dados | |
| *Actores Participantes* | | Sistema | |
| *Pre-Condiciones* | | Debe ser el turno del jugador que lanza los dados  El jugador no debe estar en el CAI | |
| *Post-Condiciones* | | *Condición final de éxito:* Jugador queda ubicado en la nueva casilla acorde al número sacado en los dados | |
| *Condiciones final de fallo:* No se mueve el jugador de su posición inicial  Ficha se mueve un número de casillas incorrectas | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 1 | Jugador lanza los dados | 2 | Evalúa que jugador no esté en el CAI |
|  |  | 3 | Mueve la ficha del jugador, según el número que haya caído en los dados |
|  |  | 4 | Actualiza tablero |

|  |  |
| --- | --- |
| *Variaciones* | 2.1:Jugador está en el CAI  2.1.1: Debe esperar a que salga del CAI para poder moverse |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU022 | *Nombre:* | Ir al CAI |
| *Nivel*  *(prioridad)* | | 3 | |
| *Garantía mínima* | | Comunicar al jugador cuando ha ido al CAI | |
| *Objetivo en Contexto:* | | Ubicar en la casilla del CAI la ficha de un determinado jugador | |
| *Actores Participantes* | | Sistema  Jugador | |
| *Pre-Condiciones* | | Jugador debe estar en su turno  Jugador cae en la casilla que lo lleva al CAI  Sacó una tarjeta que lo lleva al CAI | |
| *Post-Condiciones* | | *Condición final de éxito*: La ficha del jugador es ubicada en el CAI | |
| *Condiciones final de fallo:* Jugador no es llevado al CAI  Jugador incorrecto es llevado al CAI | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 3 | Responde a la notificación de ir a la CAI | 1 | El sistema le informa al usuario que va a ser enviado al CAI |
|  |  | 2 | Confirma información con el jugador |
|  |  | 4 | Mueve la ficha del jugador a la casilla donde se encuentra al CAI |

|  |  |
| --- | --- |
| *Variaciones* | 4.1: Jugador tiene tarjeta que lo saca al CAI  4.1.1: Sale del CAI en el siguiente turno |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU23 | *Nombre:* | Evaluar caída en casilla especial |
| *Nivel*  *(prioridad)* | | 3 | |
| *Garantía mínima* | | Notificar en qué tipo de casilla cayó el jugador | |
| *Objetivo en Contexto:* | | Verificar que tipo de casilla es la que cayó el jugador, identificar si es una casilla especial (arca comunal, servicios, propiedad, casualidad, etc) | |
| *Actores Participantes* | | Sistema | |
| *Pre-Condiciones* | | El juego debe estar inicializado  El jugador debe estar activo  Jugador está en su correspondiente turno  El jugador ya lanzó los dados y ha caído en una nueva casilla | |
| *Post-Condiciones* | | *Condición final de éxito:* Genera un reporte acerca del tipo de casilla en la que cayó el jugador, informando si esta es especial o no | |
| *Condiciones final de fallo:* No se genera ningún informe acerca del tipo de casilla en la que cayó el jugador  Genera información errónea acerca del tipo de casilla en la que está ubicada el jugador | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
|  |  | 1 | Sistema evalúa que tipo de casilla es en la que cayó el jugador |
|  |  | 2 | Informa si la casilla en la que cayó el jugador es especial |

|  |  |
| --- | --- |
| *Variaciones* |  |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU24 | *Nombre:* | Salir del CAI |
| *Nivel*  *(prioridad)* | | 3 | |
| *Garantía mínima* | | Notificar al jugador que ha salido del CAI | |
| *Objetivo en Contexto:* | | Permitir que el jugador que esté en el CAI salga de ella | |
| *Actores Participantes* | | Jugador  Sistema | |
| *Pre-Condiciones* | | El juego debe estar inicializado.  Jugador debe estar en el CAI  Para salir debe cumplir con alguna de estas condiciones:  Jugador pagó para salir  Jugador tiene tarjeta de salía del CAI  Jugador sacó pares en su turno | |
| *Post-Condiciones* | | *Condición final de éxito:* El jugador que está en el CAI sale de ella | |
| *Condiciones final de fallo:* El jugador que está en el CAI no logra salir  Jugador cumple con las condiciones de salida, pero no sale | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
|  |  | 1 | Verifica que el jugador esté en el CAI |
| 2 | Jugador realiza acción acorde para salir del CAI(lanza dados, tiene tarjeta, paga) |  |  |
|  |  | 2 | Evalúa que cumpla con alguna condición de salida |
|  |  | 3 | Permite salida del jugador |

|  |  |
| --- | --- |
| *Variaciones* | 2.1: No cumple con condición de salida  2.1.1: Termina su respectivo turno |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU025 | *Nombre:* | Finalizar juego |
| *Nivel*  *(prioridad)* | | 3 | |
| *Garantía mínima* | | Bloquear tablero | |
| *Objetivo en Contexto:* | | Dar por terminado el juego, es decir no permitir que los jugadores sigan realizando jugadas | |
| *Actores Participantes* | | Sistema | |
| *Pre-Condiciones* | | Juego debe estar activo  Un jugador se ha declarado en bancarrota | |
| *Post-Condiciones* | | *Condición final de éxito:* Lograr dar por terminado el juego evitando que los jugadores no puedan seguir realizando jugadas | |
| *Condiciones finales de fallo:* Juego no es finalizado  Jugadores continúan realizando jugadas  Jugadores tienen activo el tablero | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
|  |  | 1 | Valida si algún jugador ha quedo en bancarrota |
|  |  | 2 | Informar a los jugadores que el juego ha terminado |
|  |  | 3 | Deshabilita el tablero |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU26 | *Nombre:* | Crear partida (establecer conexión) |
| *Nivel*  *(prioridad)* | | 3 | |
| *Garantía mínima* | | Permite al jugador conectarse al servidor | |
| *Objetivo en Contexto:* | | El jugador puede crear una sesión de juego, esperando otros jugadores que se unan a ella. | |
| *Actores Participantes* | | Jugador  Sistema | |
| *Pre-Condiciones* | | Tener instalado del programa (T-Monopoly). | |
| *Post-Condiciones* | | *Condición final de éxito: El* jugador *logra crear una partida a la cual pueden unirse otros jugadores.* | |
| *Condiciones final de fallo:*  *El* jugador puede crear *una partida nueva.*  *No es posible unirse a esta partida (otros jugadores).* | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 1. | El jugador escoge la opción de crear una nueva partida. | 2. | El sistema inicia la conexión. |
| 6. | El jugador recibe información sobres sus contendores. | 3. | El sistema inicia un contador de tiempo. |
| 7. | El jugador recibe un mensaje de inicio de partida. | 4. | El sistema espera por contendores. |
|  |  | 5. | El sistema termina contador, actualiza tablero e Inicializa la partida. |

|  |  |
| --- | --- |
| *Variaciones* | 6.1: No existen contendores.  6.1.1 Se elimina la partida.  6.1.2 El jugador es informado de la eliminación de la partida. |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU27 | *Nombre:* | Eliminar Partida |
| *Objetivo en Contexto:* | | Terminar con la partida creada por un usuario, desconectar comunicaciones | |
| *Actores Participantes* | | Sistema  Jugador | |
| *Pre-Condiciones* | | Debe existir una partida para eliminarla | |
| *Post-Condiciones* | | *Condición final de éxito:* Se cierra la partida, quitando la comunicación y devolviendo al jugador a la pantalla inicial. | |
| *Condiciones final de fallo:* No se elimina la comunicación, entre los jugadores.  Se producen errores que cierran abruptamente la aplicación. | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
|  |  | 1. | Termina Time Out. |
|  |  | 2. | Cierra comunicaciones, y devuelve a pantalla inicial al jugador. |

|  |  |
| --- | --- |
| *Variaciones* | 1.1: El jugador cancela la partida. |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU028 | *Nombre:* | Actualizar estadística |
| *Nivel*  *(prioridad)* | | 3 | |
| *Garantía mínima* | | El sistema informa al jugador de las actualizaciones llevadas a cabo. | |
| *Objetivo en Contexto:* | | Al finalizar una partida se actualizan todos los atributos (partidas jugadas, partidas ganadas, partidas en banca rota) de los perfiles de los jugadores. | |
| *Actores Participantes* | | Sistema  Jugador | |
| *Pre-Condiciones* | | Verificar ganador  Finalizar juego | |
| *Post-Condiciones* | | *Condición final de éxito:* Actualizar en el perfil de cada jugador en el juego, la información correspondiente a la partida recién terminada. | |
| *Condiciones final de fallo:*  Que no se actualicen todos los atributos de cada perfil.  Que las estadísticas queden mal actualizadas. | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 4. | Envía acuse de recibido la notificación | 1 | Busca el perfil de los jugadores de la partida por ID. |
|  |  | 2 | Actualiza los datos relacionados con las estadísticas de cada jugador |
|  |  | 3 | Notifica a los jugadores |
|  |  | 5 | Recibe acuse de recibido el mensaje por parte del jugador |

|  |  |
| --- | --- |
| *Variaciones* | 4.1: No se envía acuse de recibo.  4.1.2 Termina Time Out. |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:*  *29* | CU29 | *Nombre:* | Almacenar datos partidas (propiedades, dinero, casa, hotel…) |
| *Nivel*  *(prioridad)* | | *3* | |
| *Garantía mínima* | | *El sistema es capaz de salvar datos a través del uso de archivos planos y/o binarios.* | |
| *Objetivo en Contexto:* | | Guardar información de sesión de juego. | |
| *Actores Participantes* | | Jugador  Sistema | |
| *Pre-Condiciones* | | Se ha inicializado una partida. | |
| *Post-Condiciones* | | *Condición final de éxito: El sistema almacena* (propiedades, dinero, casa, hotel…), del jugador, actualizando su respectivo perfil. | |
| *Condiciones final de fallo:*  *El Sistema no puede almacenar todos los datos del jugador* (propiedades, dinero, casa, hotel).  Los datos no son almacenados correctamente. | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 3. | El jugador recibe un mensaje de la actualización de su perfil. | 1. | El sistema actualiza los perfiles de los jugadores. |
|  |  | 2. | El sistema informa al jugador la información actualizada |
|  |  |  |  |
|  |  |  |  |

|  |  |
| --- | --- |
| *Variaciones* | 2.1: El perfil de jugador tiene una partida salvada.  2.1.1 El sistema informa al jugador si quiere sobrescribir la partida.  2.1.1.1 El jugador opta por sobrescribir la partida.  2.1.1.1.1 El sistema sobrescribe la partida, con los datos actuales  2.1.1.2 El jugador opta por no sobrescribir la partida.  2.1.1.2.1 El sistema no actualiza perfil del jugador. |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU30 | *Nombre:* | Cobro en casilla de salida |
| *Nivel*  *(prioridad)* | | *3* | |
| *Garantía mínima* | | El jugador es informado a ceca del valor que le paga el banco por pasar la casilla de salida. | |
| *Objetivo en Contexto:* | | El jugador obtiene dinero por pasar en la casilla específica. | |
| *Actores Participantes* | | Banco  Jugador | |
| *Pre-Condiciones* | | Debe ser el turno del jugador | |
| *Post-Condiciones* | | *Condición final de éxito:* Se le da determinada cantidad de dinero al jugador cuando pase por dicha casilla | |
| *Condiciones final de fallo:* No se le da el dinero al jugador.  No se actualiza el tablero. | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 1 | Jugador pasa por la casilla GO | 2 | Verifica que el jugador no acabe de salir del CAI |
| 5. | El jugador es informado del aumento de su dinero. | 3 | Asigna valor correspondiente al jugador. |
|  |  | 4. | Actualiza el tablero de juego. |

|  |  |
| --- | --- |
| *Variaciones* | 3.1: El sistema confirma que el jugador salió del CAI.  3.1.1 Se informa al jugador que no recibe dinero.  3.1.2 Se continua la secuencia del juego. |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU31 | *Nombre:* | Pagar al hippie |
| *Nivel*  *(prioridad)* | | *3* | |
| *Garantía mínima* | | Se informa al jugador de él pago al hippie. | |
| *Objetivo en Contexto:* | | EL jugador se le resta el dinero correspondiente al pago por la manilla. | |
| *Actores Participantes* | | Jugador  Sistema | |
| *Pre-Condiciones* | | Se ha llegado a una casilla de parqueadero. | |
| *Post-Condiciones* | | *Condición final de éxito: El jugador paga correctamente su multa, (Actualización de tablero, Sistema resta el monto de dinero)* | |
| *Condiciones final de fallo:*  *No se descuenta el dinero del jugador.*  *Se descuenta una cantidad errónea al dinero del jugador.* | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 2. | El jugador recibe un mensaje, pues su pareja quiere una manilla. | 1. | El sistema verifica se encuentra sobre la casilla de parqueadero. |
| 3. | El jugador realiza una acción de pago. | 4. | El sistema actualiza el historial |
|  |  | 5. | El sistema actualiza InfoJugador (ver glosario). |

|  |  |
| --- | --- |
| *Variaciones* | * 1. El jugador inicia una negociación.   3.1.1 El jugador realiza una acción de pago.   * 1. El jugador vende propiedades.   3.2.1 El jugador realiza una acción de pago. |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU32 | *Nombre:* | Consultar Perfil Jugador |
| *Nivel*  *(prioridad)* | | *3* | |
| *Garantía mínima* | | Se debe mostrar al jugador la lista de jugadores registrados. | |
| *Objetivo en Contexto:* | | Cualquier jugador dentro del juego debe poder ver los perfiles de los jugadores que se encuentran registrados. | |
| *Actores Participantes* | | Jugador  Servidor | |
| *Pre-Condiciones* | | Debe haber perfiles de usuario creados. | |
| *Post-Condiciones* | | *Condición final de éxito:* El jugador puede ver la información registrada (partidas jugadas, partidas ganadas, partidas en banca rota y nombre) por el sistema de cualquiera de los jugadores que tienen perfiles. | |
| *Condiciones final de fallo:*  El sistema no despliega información registrada de los perfiles creados,  La información de un perfil es errónea. | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 1. | El jugador escoge la opción de ver perfiles de los jugadores. | 3. | Verifica que el jugador a consultar tenga un perfil inscrito. |
| 2. | El jugador escoge el perfil del jugador que desea consultar. | 4. | El sistema despliega la información correspondiente al perfil elegido (partidas jugadas, partidas ganadas, partidas en banca rota, y nombre.) |

|  |  |
| --- | --- |
| *Variaciones* | 3.1 El perfil del jugador ha sido borrado.  3.1.1 El sistema informa al jugador que el perfil ha sido borrado recientemente.  3.1.2 El sistema actualiza el listado de jugadores inscritos. |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU33 | *Nombre:* | Asignar Propiedades Iniciales |
| *Nivel*  *(prioridad)* | | *3* | |
| *Garantía mínima* | | Se debe informar al jugador las propiedades asignadas. | |
| *Objetivo en Contexto:* | | Se debe asignar al jugador 2 propiedades aleatoriamente. | |
| *Actores Participantes* | | Jugador  Servidor | |
| *Pre-Condiciones* | | Debe haberse inicializado una partida. | |
| *Post-Condiciones* | | *Condición final de éxito: El jugador posee sus 2 propiedades.* | |
| *Condiciones final de fallo: No se reparte a todos los jugadores de la partida 2 propiedades*  *Se reparten propiedades sin libre asignación*  *Se reparten propiedades repetidas.* | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 1. | El jugador crea o se une a una partida. | 2. | Verifica que el jugador tenga un perfil inscrito. |
|  |  | 3. | Se inicializa una partida. |
|  |  | 3. | El sistema escoge aleatoriamente 2 propiedades. |
|  |  | 4. | Verifica no repetición y libre asignación (ver glosario) de las propiedades. |
| 6. | El jugador puede ver sus propiedades y obtener información sobre ellas. | 5. | El sistema asigna 2 propiedades al jugador. |

|  |  |
| --- | --- |
| *Variaciones* | 4.1 Las propiedades están repetidas o no tienen libre asignación.  4.1.1 Vuelve a 3. |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU34 | *Nombre:* | Verificar Propietario |
| *Nivel*  *(prioridad)* | | *3* | |
| *Garantía mínima* | | *El sistema permite hacer transacciones (venta, acción de pago)* | |
| *Objetivo en Contexto:* | | Cualquier transacción (venta, acción de pago) que dependa de una propiedad en el juego debe validarse quién es el dueño actual de la propiedad. | |
| *Actores Participantes* | | Sistema  Jugador | |
| *Pre-Condiciones* | | Debe haberse creado una nueva partida.  Debe estar en tránsito una partida inicializada. | |
| *Post-Condiciones* | | *Condición final de éxito: Se identifica quien posee cada una de las propiedades. (Banca, jugadores en la partida)* | |
| *Condiciones final de fallo:*  *No es posible identificar quien posee alguna propiedad.*  *Se identifica un dueño erróneamente.* | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 1. | Se inicializa una negociación. | 2. | El sistema brinda turnos de negociación. |
|  |  | 3. | El sistema verifica la correspondencia ID jugador, con ID propiedad. |
|  |  | 4. | Actualiza InfoPropiedad y TDP (ver glosario). |
| 5. | El jugador puede ver actualizada la infoPropiedad en el tablero. |  |  |

|  |  |
| --- | --- |
| *Variaciones* | 1.1: Un jugador cae en una casilla que no es de libre asignación.  1.1.1 La propiedad tiene derechos de mayor de edad. |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU35 | *Nombre:* | Escoger ficha |
| *Nivel*  *(prioridad)* | | *3* | |
| *Garantía mínima* | | *El sistema debe ofrecer una gama de posibilidades para escoger ficha.* | |
| *Objetivo en Contexto:* | | Asignarle al usuario una ficha según parámetros del jugador | |
| *Actores Participantes* | | Jugador  Sistema | |
| *Pre-Condiciones* | | Jugador debe tener un perfil. | |
| *Post-Condiciones* | | *Condición final de éxito: Se asigna al* Jugador la ficha con la que desea jugar durante toda la partida | |
| *Condiciones final de fallo:* No se le asigna la ficha al jugador.  Se asigna una ficha no escogida por el jugador. | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 2 | Selecciona ficha | 1 | Muestra los diferentes tipos de fichas |
|  |  | 3 | Valida disponibilidad de la ficha |
|  |  | 4 | Asigna la ficha la jugador |

|  |  |
| --- | --- |
| *Variaciones* | 3.1: La ficha no se encuentre disponible.  3.1.1 Se informa al jugador escoger nueva ficha.  3.1.2 Vuelve al paso 1. |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU36 | *Nombre:* | Completar vuelta al tablero |
| *Nivel*  *(prioridad)* | | 2 | |
| *Garantía mínima* | | Mostrar un mensaje al jugador de que la vuelta ha sido completada | |
| *Objetivo en Contexto:* | | Evaluar si el jugador ha realizado una vuelta completa al tablero | |
| *Actores Participantes* | | Sistema | |
| *Pre-Condiciones* | | El juego debe estar inicializado  El jugador debe estar activo  Jugador debe estar en su turno  El jugador acaba de llegar a casilla de salida  Jugador estuvo en el CAI durante esa vuelta | |
| *Post-Condiciones* | | *Condición final de éxito:* Al jugador se le asigna determinada cantidad de dinero por completar una vuelta al tablero | |
| *Condiciones finales de fallo:* No se le asigna valor correspondiente al dar una vuelta  Se asigna dinero sin que haya terminado la vuelta al juego | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 1 | Jugador finaliza turno | 2 | Evalúa que haya realizado una vuelta completa al tablero |
|  |  | 3 | Asigna determinada cantidad de dinero al jugador |
| 4 | Recibe dinero |  |  |

|  |  |
| --- | --- |
| *Variaciones* |  |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU37 | *Nombre:* | Modificar perfil |
| *Nivel*  *(prioridad)* | | 2 | |
| *Garantía mínima* | | Muestra la información del perfil del jugador para que la modifique | |
| *Objetivo en Contexto:* | | Cambiar uno o varios datos del perfil, por otros datos ingresados por el jugador | |
| *Actores Participantes* | | Jugador  Sistema | |
| *Pre-Condiciones* | | Debe existir el jugador al cual se le quiere modificar la información  Debe ser el perfil correspondiente al jugador que se quiere modificar, ningún jugador puede acceder a la información de otro jugador | |
| *Post-Condiciones* | | *Condición final de éxito: L*os datos del perfil del jugador son modificados acorde con la nueva información ingresada | |
| *Condiciones final de fallo:*  No realizar ningún cambio en la información del jugador  Cambia información del jugador erróneamente | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 1 | Jugador informa que quiere cambiar su información de perfil | 2 | Muestra la información almacenada de ese jugador |
| 3 | Jugador realiza cambios sobre su información | 4 | Pregunta si jugador está seguro de realizar estos cambios |
| 5 | Confirma que quiere realizar esos cambios | 6 | Almacena la nueva información |
|  |  | 7 | Muestra nueva información del perfil del jugador |

|  |  |
| --- | --- |
| *Variaciones* | 5.1:Jugador no confirma que quiere cambiar su información  5.1.1: No se guardan los cambios realizados |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU38 | *Nombre:* | Eliminar Perfil |
| *Nivel*  *(prioridad)* | | 2 | |
| *Garantía mínima* | | Mostrar información del usuario para eliminar | |
| *Objetivo en Contexto:* | | Eliminar determinado perfil del jugador que así lo desee | |
| *Actores Participantes* | | Jugador | |
| *Pre-Condiciones* | | Jugador debe existir  Debe ser el perfil correspondiente al jugador que se quiere eliminar, ningún jugador puede acceder a la información de otro jugador | |
| *Post-Condiciones* | | *Condición final de éxito:* El usuario es eliminado de la basa de datos del juego | |
| *Condiciones final de fallo:*  No se elimina el usuario de la base de datos | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 1 | Jugador informa que quiere eliminar su perfil | 2 | Verifica que sea su usuario |
|  |  | 3 | Muestra la información del usuario a eliminar |
|  |  | 4 | Pregunta si se quiere eliminar ese usuario |
| 5 | Confirma eliminar ese usuario | 6 | Elimina el registro de ese usuario |

|  |  |
| --- | --- |
| *Variaciones* | Paso 5: Usuario puede no confirmar la eliminación, haciendo que no se elimine ese registro |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU39 | *Nombre* | Terminar turno |
| *Nivel*  *(prioridad)* | | 2 | |
| *Garantía mínima* | | Avisar al jugador que su turno ha finalizado | |
| *Objetivo en Contexto:* | | Permitir que el jugador termine jugada y pueda ceder el turno a un contrincante. | |
| *Actores Participantes* | | Sistema | |
| *Pre-Condiciones* | | El juego esta inicializado.  El jugador ha terminado su turno  El tiempo destinado para una jugada ha terminado | |
| *Post-Condiciones* | | *Condición final de éxito:*  Se cede el turno a un contrincante. | |
| *Condiciones final de fallo:*  No es posible ceder el turno a un contrincante  No se le avisa al jugador que el turno ha terminado  No se le quita el turno al jugador, luego de cierto tiempo | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
|  |  | 1. | Evalúa que el turno del jugador haya terminado |
|  |  | 2. | Quita el turno al jugador que lo tiene |
|  |  | 3. | Asigna turno al siguiente jugador |
|  |  | 4. | Actualiza historial de jugadas |

|  |  |
| --- | --- |
| *Variaciones* |  |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU40 | *Nombre:* | Crear Perfil |
| *Nivel*  *(prioridad)* | | 2 | |
| *Garantía mínima* | | Muestra los datos necesarios para crear un perfil, con el fin que el jugador lo diligencie | |
| *Objetivo en Contexto:* | | Tener un perfil del jugador con los datos correspondientes a él | |
| *Actores Participantes* | | Jugador  Sistema | |
| *Pre-Condiciones* | | El jugador no debe existir en la base de datos del juego  jugador debe tener instalado el juego | |
| *Post-Condiciones* | | *Condición final de éxito:* Creación del perfil, con los datos ingresados | |
| *Condiciones final de fallo:* El perfil no se crea  Perfil el creado con información incorrecta | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 1 | Usuario informa que quiere crear un perfil | 2 | Muestra interfaz con los datos necesarios para crear el perfil |
| 3 | Usuario llena casillas con su información | 3 | Verifica si existe el usuario que quiere ingresar |
|  |  | 4 | Almacena información del nuevo perfil |
|  |  | 5 | Asigna ID, consecutivo acorde al que lleve el sistema |

|  |  |
| --- | --- |
| *Variaciones* | 3.1:Jugador ya existe  3.1.1: Notifica que el usuario ya existe |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU41 | *Nombre* | Abandonar Partida |
| *Nivel*  *(prioridad)* | | 2 | |
| *Garantía mínima* | | Habilita al jugador la opción de retirarse de la partida en curso | |
| *Objetivo en Contexto:* | | El jugador abandona la partida | |
| *Actores Participantes* | | Jugador  Sistema | |
| *Pre-Condiciones* | | Existe más de un jugador en el juego  Jugador se encuentra en determinado juego | |
| *Post-Condiciones* | | *Condición final de éxito:*  *El jugador abandona la partida y se permite continuar la partida.* | |
| *Condiciones finales de fallo:*  No se continúa el juego luego de la salida del jugador  Jugador no puede salir de la partida  No se entregan las posesiones al banco  No se actualiza historial | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 1. | El jugador escoge la opción de abandonar la partida. | 2. | Verifica que puede retirarse. |
| 3. | El jugador entrega posesiones | 3. | El sistema recibe propiedades, dinero y hoteles del jugador |
|  |  | 4**.** | Asigna posesiones del jugador saliente al banco. |
|  |  | 5 | Actualiza historial |

|  |  |
| --- | --- |
| *Variaciones* |  |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU42 | *Nombre:* | Iniciar Sesión con un Perfil |
| *Nivel*  *(prioridad)* | | 1 | |
| *Garantía mínima* | | Permitir al usuario ingresar los datos correspondientes a su perfil | |
| *Objetivo en Contexto:* | | Ingresa a una sesión por medio de un identificador único para cada jugador (id, nombre) | |
| *Actores Participantes* | | Jugador  Sistema | |
| *Pre-Condiciones* | | El usuario debe estar registrado  Los valores deben estar almacenada | |
| *Post-Condiciones* | | *Condición final de éxito:* El usuario ingresa a la sesión sin ningún problema | |
| *Condiciones finales de fallo:*  Usuario no puede acceder a una sesión | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 1 | Jugador accede a opción para ingresar al sistema | 3 | Valida que la información exista y sea correcta |
| 2 | Ingresa información de su perfil |  |  |
|  |  | 4 | Habilita entrada a la sesión al jugador con ese perfil |

|  |  |
| --- | --- |
| *Variaciones* | 3.1: Información ingresada no es correcta  3.1.1: Se pide al jugador que rectifique su información |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU043 | *Nombre:* | Mostrar estadísticas |
| *Nivel*  *(prioridad)* | | 1 | |
| *Garantía mínima* | | Habilitar al jugador la opción de ver estadísticas del juego | |
| *Objetivo en Contexto:* | | Le permite conocer a cada jugador sobre las estadísticas de cualquier otro jugador | |
| *Actores Participantes* | | Sistema  Jugador | |
| *Pre-Condiciones* | | Tener perfil activado  Jugador debe tener una sesión iniciada  Jugador debe haber terminado el juego  Jugador no ha iniciado partida | |
| *Post-Condiciones* | | *Condición final de éxito:* Mostrar las estadísticas sobre partidas anteriores del jugador seleccionado | |
| *Condiciones finales de fallo:* Las estadísticas que se muestren no sean del jugador deseado  Información errónea de las estadísticas | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 1 | Selecciona jugador | 2 | Captura id del jugador seleccionado |
|  |  | 3 | Busca estadísticas por el id del jugador seleccionado |
| 7 | Envía confirmación de información recibida | 4 | Muestra las estadísticas del jugador seleccionado |
|  |  | 5 | Espera que usuario confirme que ha visto la estadística |

|  |  |
| --- | --- |
| *Variaciones* | 7.1: Jugador no confirma que recibió las estadísticas  7.1.1: Reduce tiempo de jugada del jugador |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU44 | *Nombre:* | Mostrar mensajes al jugador |
| *Nivel*  *(prioridad)* | | 1 | |
| *Garantía mínima* | | Bloquea tablero del jugador cuando se le quiera mostrar un mensaje | |
| *Objetivo en Contexto:* | | Informar al jugador acerca de su estado en el juego, propiedades que obtuvo en ese turno, cover que debe cancelar , dinero asignado por el banco, etc | |
| *Actores Participantes* | | Sistema | |
| *Pre-Condiciones* | | El jugador debe estar conectado al juego  Aparece luego de determinada actividad que requiera ser informada (envío al CAI, pago de cover, compra de propiedad, etc)  Jugador ha finalizado turno  Jugador ha finalizado jugada | |
| *Post-Condiciones* | | *Condición final de éxito:* Se muestra al usuario determinado estado de su dinero, propiedades ó demás situaciones que le suceden a medida que avanza en el juego | |
| *Condiciones finales de fallo:* No se le muestra el mensaje al jugador acerca de determinado estado o situación ocurrida en el desarrollo del juego.  Se le muestra un mensaje con información incorrecta acerca del estado de las posesiones del jugador | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
|  |  | 1 | Muestra una ventana con la información del estado del jugador luego de esa partida |
| 2 | Jugador acepta la información que se le está mostrando | 3 | Continua con el juego |

|  |  |
| --- | --- |
| *Variaciones* |  |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU45 | *Nombre:* | Validar Casilla |
| *Nivel*  *(prioridad)* | | 1 | |
| *Garantía mínima* | | Evalúa casilla en la que cae el jugador durante su turno | |
| *Objetivo en Contexto:* | | Identificar qué tipo es la casilla, si es propiedad, corresponde a una casilla de multa, ir al CAI, pagos de servicios, arca comunal, causalidad, etc. | |
| *Actores Participantes* | | Sistema | |
| *Pre-Condiciones* | | Juego debe estar inicializado  Jugador debe haber lanzado dados  Jugador debe estar ubicado en una nueva casilla  Casilla en la que está el jugador debe ser válida (estar dentro del tablero) | |
| *Post-Condiciones* | | *Condición final de éxito:* Genera informe con el tipo de la casilla en la que está ubicado el jugador | |
| *Condiciones finales de fallo:* No identifica la casilla en la que está el jugador  No reconoce el tipo de casilla en la que se encuentra ubicado el jugador  Envía un tipo de casilla erróneo  No genera reporte de la casilla en la que está ubicado el jugador | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
|  |  | 1 | Evalúa que tipo es la casilla en la que está el jugador |
|  |  | 2 | Genera reporte acorde al tipo de casilla obtenido |

|  |  |
| --- | --- |
| *Variaciones* |  |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU46 | *Nombre:* | Lanzar los dados |
| *Nivel* | | x | |
| *Garantía mínima* | | Mostrar los dados para ser lanzados | |
| *Objetivo en Contexto:* | | Jugador lanza los dados en su turno, obteniendo un número de estos, indicando el movimiento que debe realizar | |
| *Actores Participantes* | | Jugador | |
| *Pre-Condiciones* | | Debe ser el turno correspondiente al jugador | |
| *Post-Condiciones* | | *Condición final de éxito:* Jugador obtiene un número como resultado del lanzamiento de los dados, el cual corresponde al número de casillas que debe moverse en el tablero | |
| *Condiciones final de fallo:* Lanzamiento de los dados no se realiza, es decir , obtiene un número de casillas requerido para moverse | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 1 | Jugador notifica que lanza los dados | 2 | Muestra lanzamiento de los dados en pantalla |
|  |  | 3 | Muestra el numero que dieron los dados |

|  |  |
| --- | --- |
| *Variaciones* | * 1. Demora en la notificación por parte del jugador   1.1.1 Inicio del Time – Out |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU47 | *Nombre:* | Asignar propiedad a un jugador |
| *Nivel* | | x | |
| *Garantía mínima* | | Identificar la propiedad que se va a asignar | |
| *Objetivo en Contexto:* | | Entregar una propiedad al jugador luego que este ha cancelado el dinero necesario para obtenerla | |
| *Actores Participantes* | | Jugador | |
| *Pre-Condiciones* | | Jugador debe estar en su turno para realiza dicha compra ó haber ganado en la subasta para obtenerla  Luego de que el jugador ha realizado una compra de dicha propiedad al banco o a otro jugador | |
| *Post-Condiciones* | | *Condición final de éxito:* Al jugador se le asigna la propiedad que ha cancelado | |
| *Condiciones final de fallo:* No se le asigna la propiedad al jugador | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 1 | Jugador decide comprar determinada propiedad | 2 | Muestra cantidad correspondiente a la cantidad por la que oferta el jugador |
| 3 | Cancela cantidad correspondiente a esa propiedad | 4 | Asigna propiedad a el jugador que la canceló |

|  |  |
| --- | --- |
| *Variaciones* | 1.1 En caso de que el jugador no compre la propiedad  1.1.1 Se continua con el juego normalmente  3.1 En caso de que el jugador no cuente con la cantidad de dinero para comprar la propiedad  3.1.1 Se le informa al jugador y no se le asigna la propiedad |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU48 | *Nombre:* | Asignar trago a jugador |
| *Nivel* | | x | |
| *Garantía mínima* | | Que se identifica el trago que va a ser asignado al jugador | |
| *Objetivo en Contexto:* | | Asignarle el trago que el jugador ha pagado | |
| *Actores Participantes* | | Jugador | |
| *Pre-Condiciones* | | Debe estar el jugador en su turno, para realizar la compra de tragos  Jugador debe tener la cantidad necesaria para realizar la compra de los tragos que está solicitando  Jugador no debe exceder en tragos, es decir no debe tener más de cuatro cervezas ó una botella de Sello azul por propiedad | |
| *Post-Condiciones* | | *Condición final de éxito:* Se le asigna al jugador el trago que está pagando | |
| *Condiciones final de fallo:* No se le asigna el trago comprado al jugador | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 1 | Jugador solicita determinado número de tragos | 2 | Evalúa que se cumplan las reglas del juego, que no exceda número de tragos por propiedad |
| 3 | Cancela los tragos que solicitó | 4 | asigna tragos solicitados al jugador |

|  |  |
| --- | --- |
| *Variaciones* | 2.1 En caso de que no se cumplan las reglas mínimas  2.1.1 Se cancela el pedido y se continua con el juego |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU49 | *Nombre:* | Comprar trago a jugador |
| *Nivel* | | x | |
| *Garantía mínima* | | Identificar el jugador al que se va a vender el trago | |
| *Objetivo en Contexto:* | | El banco compra el trago ofrecido por el jugador | |
| *Actores Participantes* | | Banco | |
| *Pre-Condiciones* | | Jugador debe tener trago disponible para venderle al banco | |
| *Post-Condiciones* | | *Condiciones final de éxito:* Jugador vende su trago al banco, por un precio acordado | |
| *Condición final de fallo:*  Jugador no realiza la venta del trago al banco | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 1 | Jugador ofrece trago al banco | 2 | Cancela la cantidad solicitada para comprar el trago |
| 3 | Recibe dinero | 4 | Se le asigna trago al banco |

|  |  |
| --- | --- |
| *Variaciones* | 4.1 En caso de que se asigne mal el trago  4.1.1 Verificar el jugador comprador del trago |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU51 | *Nombre:* | Declarar Banca Rota |
| *Nivel:* | | x | |
| *Garantía mínima:* | | Habilitar el botón de declaración de banca rota | |
| *Objetivo en Contexto:* | | Un jugador se declara en Banca rota | |
| *Actores Participantes* | | Jugador | |
| *Pre-Condiciones* | | El jugador no tiene propiedades suficientes o dinero suficiente para cumplir un pago. | |
| *Post-Condiciones* | | *Condiciones final de éxito:* El jugador es declarado en banca rota. | |
| *Condición final de fallo:* El jugador no es declarado en banca rota. | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 1 | Oprime el botón de declararse en bancarrota | 2 | Pregunta confirmación del usuario |
| 3 | Confirma petición de declaración | 4 | Verifica posesiones del jugador |
|  |  | 5 | Informa sobre bancarrota de un jugador a todos los demás jugadores |

|  |  |
| --- | --- |
| *Variaciones* | 3.1 En caso de que el jugador no confirme la petición  3.1.1 Se continua normalmente con el juego  4.1 En caso de que todavía tenga posesiones se le notifica al jugador |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU52 | *Nombre:* | Deshabilitar Tablero |
| *Nivel:* | | x | |
| *Garantía mínima:* | | Bloquear las jugadas de los jugadores | |
| *Objetivo en Contexto:* | | Impedir que los jugadores sigan haciendo jugadas durante el turno de otro ó al ser declarados en banca rota | |
| *Actores Participantes* | | Sistema | |
| *Pre-Condiciones* | | Hay un jugador en banca Rota.  Jugador no tiene el turno correspondiente | |
| *Post-Condiciones* | | *Condiciones final de éxito*: Los jugadores no pueden continuar haciendo jugadas hasta que sea su turno | |
| *Condición final de fallo*: Los jugadores pueden continuar haciendo alguna jugada. | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
|  |  | 1 | Evalúa que jugador tiene el turno |
|  |  | 2 | Habilita tablero al jugador que tiene turno |

|  |  |
| --- | --- |
| *Variaciones* |  |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU53 | *Nombre* | Acción de Pago |
| *Nivel:* | | x | |
| *Garantía mínima:* | | Descontar el dinero al jugador que paga | |
| *Objetivo en Contexto:* | | El jugador realiza un pago correctamente. | |
| *Actores Participantes* | | Jugador  Sistema | |
| *Pre-Condiciones* | | El juego esta inicializado.  El jugador puede pagar su deuda con dinero o propiedades. | |
| *Post-Condiciones* | | *Condiciones final de éxito:*  *Se cancela el cobro del* jugador *correctamente.* | |
| *Condición final de fallo:*  *No es posible cancelar el cobro del* correctamente. | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 1. | El jugador cae sobre propiedades ajenas que pueden cobrar. | 2. | El sistema verifica que se puede cobrar y el valor por el cuál debe hacerse. |
| 3. | El jugador cubre el cobro generado. | 4. | El sistema verifica para quien es el dinero correspondiente |
| 5. | El jugador recibe información sobre el gasto realizado. | 6. | El sistema asigna el dinero, suma y descuenta a las partes respectivas. |

|  |  |
| --- | --- |
| *Variaciones* | * 1. El jugador inicia una negociación.   2. El jugador vende propiedades. |

|  |  |  |  |
| --- | --- | --- | --- |
| *Id Caso de Uso:* | CU54 | *Nombre:* | Vender bien raíz a banco |
| *Objetivo en Contexto:* | | Vender un bien raíz al Banco (propiedades, casas, hoteles) | |
| *Actores Participantes* | | Jugador  Banco. | |
| *Pre-Condiciones* | | El jugador está en su turno correspondiente.  El jugador no ha lanzado los dados. | |
| *Post-Condiciones* | | *Condiciones final de éxito:*  *El bien raíz es vendido correctamente a un segundo* jugador2*, o a la banca respectivamente.* | |
| *Condición final de fallo:*  *El bien raíz no puede ser vendido a un segundo* jugador2*, o a la banca respectivamente.* | |

|  |  |  |  |
| --- | --- | --- | --- |
| *Flujo básico de éxito* | | | |
|  | *Actor* |  | *Sistema* |
| 1. | El jugador inicia negociación. |  |  |
| 3. | El jugador escoge vender propiedad. | 2. | El sistema avisa a los demás jugadores el tipo de negociación abierta. |
| 4. | El jugador escoge el valor, y la forma de venta. | 5. | El sistema envía a los demás jugadores la oferta. |
| 6. | El jugador escoge una propuesta o rechaza su oferta. | 7. | El sistema realiza la acción de pago, y verifica propiedades. |

|  |  |
| --- | --- |
| *Variaciones* | 4.1 El jugador vende al Banco.  4.2 El banco decide el precio.  5. Salta este paso. |