



Universidad
Zaragoza

Cluedo Unizar

Práctica Ingeniería de Software

Samuel Felipe

3º GII

Félix Serna Fortea

Contenido

1. INTRODUCCION	4
1.1. Propósito.....	4
1.2. Ámbito.....	4
1.3. Referencias.....	5
1.4. Descripción del documento.....	5
2. DESCRIPCION GENERAL.....	6
2.1. Perspectiva del producto.....	6
2.1.1. Interfaces del sistema.....	6
2.1.2. Interfaces de usuario.....	6
2.1.3. Interfaces Hardware	6
2.1.4. Interfaces Software.....	6
2.1.5. Interfaces de comunicaciones.....	6
2.1.6. Restricciones de Memoria y Almacenamiento.....	7
2.1.7. Modos de Operación.....	7
2.1.8. Necesidades de infraestructura.....	7
2.2. Funcionalidad del Producto	8
2.2 Características de los usuarios	11
2.3 Restricciones	11
2.3.1 Regulación normativa.....	11
2.3.2 Limitaciones de hardware.....	11
2.3.3 Interfaces con otras aplicaciones.....	11
2.3.4 Operaciones paralelas	12
2.3.5 Auditoría.....	12
2.3.6 Control.....	12
2.3.7 Lenguajes de programación	12
2.3.8 Protocolos de comunicaciones	12
2.3.9 Fiabilidad	12
2.3.10 Criticidad	13
2.3.11 Seguridad	13
2.4 Asunciones y dependencias.....	13
2.5 Requisitos posibles en el futuro.....	14
3. REQUISITOS ESPECIFICOS	14

3.1	Interfaces externas.....	14
3.1.1.	Interfaces de usuario.....	15
3.1.2.	Interfaces hardware	15
3.1.3.	Interfaces software	15
3.1.4.	Interfaces de comunicaciones.....	15
3.2	Características del sistema.....	15
3.2.1	Inicio y cierre del sistema.....	15
3.2.2	Gestión de cuentas.....	16
3.2.3	Administración de partidas	17
3.2.4	Interacción con el tablero y elementos de la partida	18
3.3	Requisitos de prestaciones	20
3.4	Requisitos de persistencia	21
3.5	Requisitos de diseño	21
3.6	Atributos del sistema	21
3.6.1	Fiabilidad	21
3.6.2	Disponibilidad.....	21
3.6.3	Seguridad	21
3.6.4	Mantenibilidad	21
3.6.5	Escalabilidad	22
3.6.6	Portabilidad.....	22
4.	DISEÑO	22
4.1	Diagramas de casos de uso.....	22
4.2	Diagrama de clases	25
4.3	Diagramas de actividades	25
4.4	Diagramas de secuencia.....	29

1. INTRODUCCION

1.1. Propósito

Este documento presenta el desarrollo de las fase de análisis y diseño de la aplicación Cluedo UNIZAR, para la realización de la práctica de ingeniería de Software de la Escuela Universitaria Politécnica de Teruel

1.2. Ámbito

Cluedo UNIZAR es un videojuego multijugador basado en el juego de mesa Cluedo, ambientado en el contexto de la universidad de Zaragoza, por lo que primeramente se ha de entender cómo se juega al juego original de mesa Cluedo.

Se ha producido un asesinato y los jugadores deberán hacerse 3 preguntas durante toda la partida, quien ha asesinado, con que ha asesinado y donde ha asesinado. Para ello deberán utilizar la lógica, e ir recopilando información a lo largo de la partida. El juego tiene diferentes componentes:

- **Cartas:**

- **Sospechosos:** representan los posibles culpables del crimen, y cada jugador al entrar a la partida es uno de los sospechosos.
- **Armas:** El asesinato se ha producido con un arma concreta, por lo que las cartas de armas son los posibles objetos con los que se ha producido el asesinato
- **Lugares:** son aquellos sitios donde se ha producido el asesinato, también son los sitios del mapa mientras se juega.

- **Tablero:**

Es el entorno en el que se produce la partida, y esta dividido en casillas. En el tablero hay repartidas diferentes zonas específicas como pueden ser habitaciones o lugares concretos, los cuales coinciden con los de las cartas, los jugadores por lo tanto deberán moverse por el tablero para explorar el entorno y recoger pistas.

Al principio del juego, se barajan estos 3 grupos de cartas, y se elige una de cada grupo al azar, estas representaran quien, donde y con que se ha producido el asesinato, por lo que se han de guardar hasta el final de la partida. El resto de las cartas se reparten a los jugadores equitativamente, las cuales serán privadas para cada jugador. Si sobran cartas, se mostrarán a todos los jugadores. Posteriormente se elige al azar el orden de turnos de cada jugador, en cada turno el jugador empezará en una casilla inicio, y deberá tirar unos dados en los que el resultado será la distancia que puede recorrer en el tablero desde su posición actual. Los jugadores deben de ir a lugares concretos del tablero, que coinciden con las cartas de lugares que se han repartido. En estos lugares, los jugadores harán una acusación por turno, siendo el lugar el que se encuentra, quien y con que se ha producido el

asesinato. Si alguno del resto de jugadores no tiene ninguna carta de las que se ha producido la acusación no hará nada. Si tiene una carta, entonces deberá enseñársela al jugador que ha hecho la acusación, y si tiene varias cartas coincidentes deberá elegir una de ellas. De esta manera el jugador tendrá durante toda la partida una tabla en la que se irán tachando las posibles combinaciones de asesinatos, en función de las acusaciones que él va haciendo o las cartas que tiene. De esta manera los jugadores deberán jugar y hacer acusaciones de manera estratégica para ir descartando posibles asesinatos. En cualquier momento, los jugadores pueden hacer una acusación definitiva, en la que se comprueba las 3 cartas que al principio han sido seleccionadas. Si coinciden, entonces el jugador gana la partida, y si no coinciden el jugador queda eliminado de la partida. Por lo que los jugadores deberán de tratar tener claro cómo se ha producido el asesinato para poder hacer la acusación final.

La aplicación por tanto permitirá jugar al Cluedo, dejando al jugador iniciar partidas privadas o partidas online. Estas partidas son de 2 a 8 jugadores, y si son privadas solo se podrá acceder a la partida mediante un código que ha sido generado al crearla. Las partidas públicas en contraposición son creadas automáticamente, y los jugadores se meten en partidas con jugadores de su mismo nivel. Cada partida pública que se juega se obtienen puntos dependiendo de cómo se ha jugado la misma, con estos puntos es con los que se sube de nivel.

1.3. Referencias

- "IEEE Recommended Practice for Software Requirements Specifications," en IEEE Std 830-1998, vol., no., pp.1-40, 20 Oct. 1998. Disponible en: <https://ieeexplore.ieee.org/document/720574>
- "RQ-Stockk_borrador_2024-09-26". Documento obtenido del campus virtual de la Universidad de Zaragoza, dentro del apartado de recursos de la asignatura de Ingeniería del Software.

1.4. Descripción del documento

En el capítulo 2 se realiza un desglose general de la funcionalidad de la aplicación Cluedo UNIZAR. Se presenta un diagrama UML con los casos de uso, mostrando los actores que interactúan, así como sus funcionalidades en diferentes escenarios. Se identificarán los requisitos genéricos de usuario, tipos de usuario y algunas restricciones que deberán ser consideradas.

En el capítulo 3 se desarrollará lo explicado en el capítulo 2, definiendo todos los requisitos del sistema.

En el capítulo 4 se presentará un diseño general de alto nivel de la aplicación.

2. DESCRIPCION GENERAL

2.1. Perspectiva del producto

2.1.1. Interfaces del sistema

- No hay.

2.1.2. Interfaces de usuario

- La interfaz será el principal punto de entrada de los usuarios para poder utilizar la aplicación. En esta interfaz se permitirá al usuario buscar partidas online y crear partidas privadas. En las partidas el usuario tendrá a disposición una vista del tablero con todos los jugadores de la partida en su posición actual, y una tabla con las cartas descartadas como parte del asesinato.

2.1.3. Interfaces Hardware

- No hay.

2.1.4. Interfaces Software

- Cluedo UNIZAR deberá conectarse a un servicio de base de datos para almacenar y gestionar información sobre usuarios, partidas y estadísticas del juego.
- Se integrará con servicios externos para el inicio de sesión, como Google OAuth.
- Se utilizará API REST para gestionar la comunicación entre el cliente y el servidor de la aplicación.

2.1.5. Interfaces de comunicaciones

- Cluedo UNIZAR utilizara protocolos habituales en las aplicaciones web como son las comunicaciones HTTP/HTTPS para la interacción entre usuarios y peticiones a las APIs.
- La aplicación contara con soporte para WebSocket para la sincronización en tiempo real entre los participantes en las partidas.

2.1.6. Restricciones de Memoria y Almacenamiento

- No las hay o no son significativas dado el entorno previsto.

2.1.7. Modos de Operación

Pueden distinguirse cuatro modos de Operación en Cluedo UNIZAR, dependiendo el contexto y el tipo de usuario en cada caso:

- Usuario administrador mediante interfaz web
- Usuario estándar mediante interfaz web
- Usuario administrador mediante interfaz externa (REST)
- Usuario estándar mediante interfaz externa (REST)

2.1.8. Necesidades de infraestructura

- Servidor de aplicaciones web considerando la posibilidad de alta concurrencia durante eventos masivos (muchos jugadores activos)
- Infraestructura de alojamiento propia en los servidores de UNIZAR o infraestructura en la nube (AWS, Azure, etc.)
- Soporte en tiempo real entre jugadores en partidas con WebSocket Server o soluciones de mensajería como Firebase.

2.2. Funcionalidad del Producto

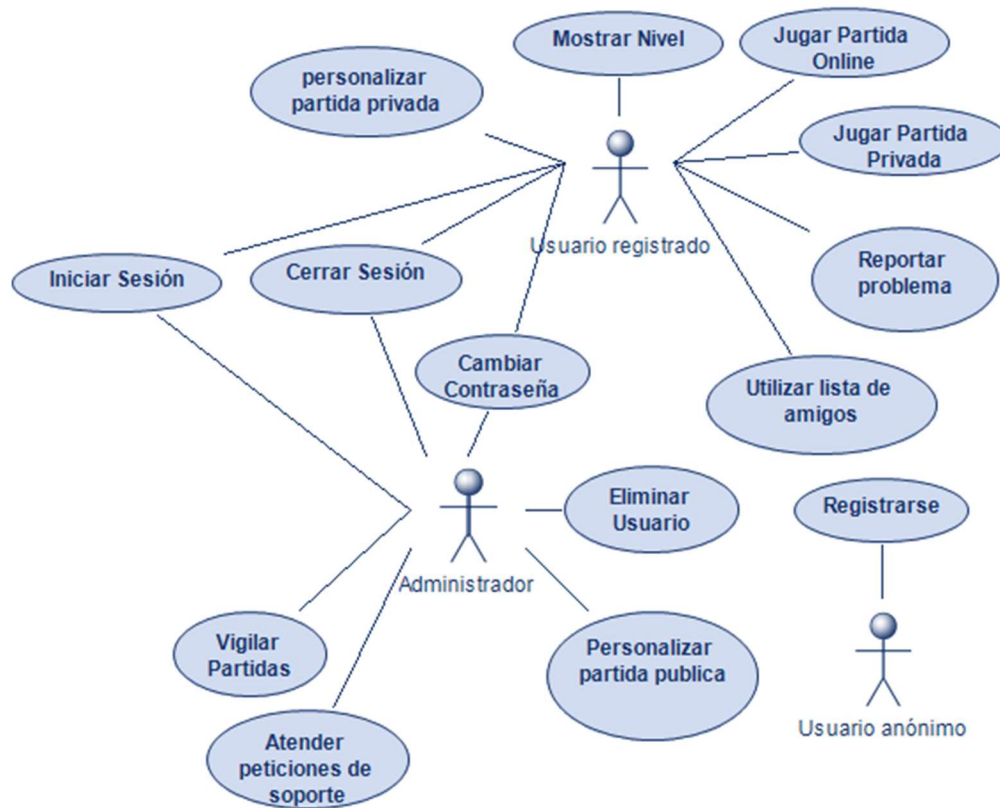


Ilustración 1: caso de uso general

En este apartado se va a considerar las acciones principales desde el punto de vista de un usuario sin registrar, un usuario administrador, y un usuario registrado, profundizando posteriormente en el funcionamiento de la aplicación mientras hay una partida en curso, de momento se explicará a continuación las acciones permitidas antes de empezar cualquier partida.

Usuario anónimo:

- Un usuario no registrado solo podrá únicamente crear una cuenta y registrarse.

Usuario Administrador:

- Podrá eliminar cualquier cuenta de usuario, eliminando así todos los datos asociados incluidos niveles, estadísticas y listas de amigos.
- Atenderá peticiones de soporte enviadas por los usuarios.
- Sera capaz de poder crear, modificar y eliminar diseños de partidas específicas, incluyendo las cartas y su número, el entorno y los

personajes. Estas partidas podrán ser jugadas por los usuarios registrados.

- El administrador podrá supervisar partidas en curso para así detectar posibles problemas técnicos o conductas inapropiadas.
- Podrá evidentemente realizar acciones básicas como iniciar sesión, cerrar sesión o cambiar su contraseña.

Usuario registrado:

- Se permitirá iniciar sesión a un usuario con sus credenciales previamente registradas.
- Un usuario registrado podrá también desconectarse de su cuenta.
- También podrá modificar su contraseña.
- Podrá consultar en cualquier momento su lista de amigos, en la cual puede añadir y eliminar a usuarios de la lista.
- El sistema facilitara al usuario el nivel actual del usuario basado en los puntos acumulados en partidas públicas anteriores.
- El usuario podrá iniciar partidas públicas, en las cuales jugará con usuarios con un nivel igual o similar. Igualmente se jugará la partida que quiera (con tablero, habitaciones y cartas diferentes) disponiendo de una serie de diseños establecidos por los administradores.
- El usuario registrado podrá jugar partidas privadas, pudiendo crear una partida privada en la que le será facilitado un código único que podrá compartir, o también podrá invitar a amigos de su lista de amigos. También podrá unirse a partidas privadas que le inviten, o introduciendo el código correspondiente. Podrá también crear sus propios diseños para partidas privadas únicamente.

Para este apartado se va a definir las acciones principales de los personajes en el caso de una partida en curso.

Usuario Administrador:

- Un usuario administrador le será permitido consultar el estado actual de la partida, incluyendo datos que pueden ver todos los usuarios como la posición de cada jugador en el tablero o cada acusación de cada jugador, como también las cartas que tiene cada jugador y las cartas que ha ido descartando.

- También podrá expulsar a cualquier jugador registrado si el administrador lo considera oportuno.
- Podrá interactuar con un chat de texto compartido con todos los integrantes y administradores de la partida, en el que enviar mensajes y leerlos.

Usuario registrado:

- El usuario registrado (no administrador) solo podrá hacer las acciones marcadas por un número solo en el caso de que sea su turno, en orden de los números y una única vez por turno, el resto de las acciones las podrá hacer en cualquier momento de la partida.
- Podrá interactuar con un chat de texto, pudiendo leer o enviar mensajes por el mismo chat compartido con todos los jugadores de la partida y posibles administradores supervisando la misma.
- Podrá consultar en cualquier momento la posición en el tablero de todos los jugadores, las cartas descartadas como posibles en el asesinato y observar acusaciones que van realizando los jugadores en cada momento, y las cartas que le han sido repartidas, pero en ningún caso podrá ver las cartas del resto de jugadores ni las cartas descartadas como posibles implicadas en el asesinato. Hay que destacar que las cartas descartadas como posibles en el asesinato son unos datos que se irán actualizando conforme el paso de la partida y las acciones.
- Cuando sea el turno del jugador, podrá hacer las siguientes acciones en orden:
 - Tirar los dados y dependiendo del resultado final, moverse a una posición que necesite un recorrido menor o igual al resultado de los dados, si lo prefiere puede pasar el turno al siguiente jugador. En esta ocasión también puede hacer una acusación final si el jugador lo prefiere, si dicha acusación coincide con las 3 cartas, entonces el jugador gana la partida, sino será eliminado.
 - Si se mueve al final y se encuentra en un lugar concreto (los correspondientes a las cartas) podrá hacer una acusación, solo en ese caso podrá hacer una suposición de un asesinato, mencionando donde se ha producido el asesinato (no pudiendo elegir otro lugar salvo en el que se encuentra) quien lo ha producido y con qué. Para el resto de los jugadores, si no tienen ninguna carta que coincida con las de la acusación no harán nada, si una carta coincide han de enseñársela al acusador y solo al acusador, y si más de una coincide, podrán elegir que carta enseñar, enseñando obligatoriamente una carta y solo una al acusador. De esta manera, el acusador podrá descartar posibles

asesinatos, ya que las cartas que tienen el resto de los jugadores no son las del asesinato.

- Dependiendo del desarrollo de la partida, el jugador obtendrá una serie de puntuaje premiando acciones favorables como descubrimiento de cartas de los rivales, o ganar la partida. Por supuesto se penalizarán acciones desfavorables como hacer una acusación definitiva y fallar, o tardar demasiado en su turno. Cada jugador tiene 1 minuto para completar su turno, al terminar el minuto si no se han terminado las acciones entonces se pasará el turno, y si un jugador se pasa 3 veces de ese minuto, será expulsado de la partida. Esto para que, siendo un juego por turnos, se elimine el entorpecimiento en la medida de lo posible.

2.2 Características de los usuarios

Los usuarios pueden interactuar con la aplicación por medio de la interfaz web, además de por medio de aplicaciones externas mediante API REST para información que no es en tiempo real como inicio de sesión o consulta de nivel, o con WebSockets para notificaciones en tiempo real en una partida.

2.3 Restricciones

2.3.1 Regulación normativa

La aplicación Cluedo UNIZAR no está sujeta a regulaciones normativas específicas más allá de las leyes de privacidad de datos (como el RGPD en la Unión Europea) para la gestión de datos personales de los usuarios incluidos las contraseñas.

2.3.2 Limitaciones de hardware

El juego está diseñado para ejecutarse en dispositivos con capacidades estándar, como ordenadores personales, dispositivos móviles y tabletas, con requisitos mínimos:

- Navegadores modernos (Google Chrome, Firefox, Safari, Edge).
- Conexión a internet estable para partidas en tiempo real. No se requiere hardware especializado.

2.3.3 Interfaces con otras aplicaciones

La aplicación no integra directamente con otros sistemas o plataformas externas. Sin embargo, se ofrecen API REST para la interacción con

aplicaciones externas (por ejemplo, aplicaciones móviles o sistemas de estadísticas).

2.3.4 Operaciones paralelas

El sistema permite operaciones concurrentes en partidas múltiples. Cada partida tiene su propio contexto y estado, gestionado de manera independiente por el servidor para evitar interferencias entre usuarios.

2.3.5 Auditoría

Se llevará un registro básico de acciones críticas durante las partidas, como movimientos, acusaciones y desconexiones en la partida que quedaran guardados.

2.3.6 Control

Los administradores del sistema tienen herramientas para supervisar las partidas activas, gestionar usuarios y resolver problemas reportados. No se consideran sistemas automáticos de control externo.

2.3.7 Lenguajes de programación

La implementación de la aplicación utiliza tecnologías web modernas:

- Backend: Node.js o Python (con frameworks como Express.js o Django).
- Frontend: JavaScript/TypeScript con frameworks como React o Astro
- Opcionalmente, tecnologías específicas pueden adaptarse según los requisitos del proyecto.

2.3.8 Protocolos de comunicaciones

Se utilizan protocolos estándar para garantizar la interoperabilidad:

- HTTP/HTTPS para comunicación RESTful.
- WebSocket para comunicación en tiempo real durante las partidas. Todos los datos sensibles se cifran en tránsito mediante HTTPS.

2.3.9 Fiabilidad

El sistema está diseñado para soportar múltiples partidas simultáneamente. En caso de fallos en la red o desconexiones de jugadores, se intentará reanudar automáticamente la conexión. Si el servidor falla, las partidas en curso podrían interrumpirse.

2.3.10 Criticidad

El sistema no es crítico, ya que no implica riesgos para la vida humana ni operaciones sensibles. Sin embargo, la experiencia del usuario se prioriza para evitar interrupciones innecesarias durante el juego. En el caso de los turnos, se permite como máximo 1 minuto por turno pasando el turno si el tiempo excede, y expulsando al jugador si repite esta conducta repetidas veces.

2.3.11 Seguridad

La aplicación implementa medidas de seguridad para proteger los datos y evitar accesos no autorizados:

- Autenticación mediante tokens JWT.
- Protección contra ataques comunes (ej. inyección SQL, CSRF).
- Cifrado de datos en tránsito y, cuando corresponda, en reposo.
- Gestión adecuada de permisos de usuario y administrador.

2.4 Asunciones y dependencias

En el diseño y funcionamiento de la aplicación Cluedo UNIZAR, se realizan las siguientes asunciones para simplificar y agilizar el desarrollo del juego, aunque no reflejen necesariamente la realidad de un juego de mesa tradicional:

- Se asume que los jugadores siempre juegan de manera honesta y respetan las reglas del juego en cuanto a movimientos y acusaciones.
- Se asume que un jugador puede ser expulsado automáticamente por inactividad o comportamiento disruptivo según las reglas definidas (ej. exceder tiempo en turnos repetidamente).
- Se asume que las partidas privadas cuentan con un sistema de invitaciones robusto, y solo los usuarios con el código correcto podrán unirse a estas partidas.
- La representación digital de las cartas, el tablero y la tabla de cartas descartadas se simplifica para adaptarse al entorno digital, reduciendo los componentes visuales o físicos necesarios en el juego de mesa original.

Estas asunciones permiten que el juego sea ágil, fácil de implementar y accesible para los jugadores a través de la plataforma digital.

2.5 Requisitos posibles en el futuro

Para mejorar la experiencia de los jugadores y acercar el juego digital a una experiencia más rica y cercana al juego original de Cluedo, se consideran los siguientes requisitos posibles a implementar en el futuro:

- Introducción de personalización avanzada de partidas:
 - Configurar reglas específicas (ej. tiempo por turno, cantidad máxima de turnos).
 - Elección de temas personalizados para las cartas y el tablero (ambientación diferente al contexto de la Universidad de Zaragoza).
- Implementación de un sistema de inteligencia artificial para jugadores:
 - Agregar bots para que los jugadores puedan participar en partidas en local sin depender de otros usuarios conectados.
- Mejoras en el sistema de reportes y sanciones:
 - Introducir un sistema de revisión y apelación para jugadores expulsados de partidas debido a comportamiento disruptivo.
- Ampliar las funcionalidades de análisis:
 - Proveer estadísticas detalladas sobre el desempeño de los jugadores, incluyendo datos sobre el tiempo promedio por turno, acusaciones realizadas y precisión en las acusaciones finales.
- Incorporar un sistema de progresión más elaborado:
 - Implementar niveles, logros y recompensas para fomentar la competitividad y la participación recurrente.
- Expansión del modelo de interacción:
 - Habilitar chat por voz en partidas privadas y públicas para mejorar la comunicación y el dinamismo del juego.
 - Introducir pistas dinámicas o eventos aleatorios durante las partidas que modifiquen el curso del juego para hacerlo más interesante.

3. REQUISITOS ESPECIFICOS

3.1 Interfaces externas

3.1.1. Interfaces de usuario

- **RQ1.** La aplicación deberá ser accesible desde navegadores web modernos que soporten estándares como HTML5 y CSS3.
- **RQ2.** La interacción con el usuario se realizará mediante una interfaz intuitiva basada en formularios, listas y gráficos, siendo adaptable a cualquier tamaño de pantalla.
- **RQ3.** La aplicación deberá soportar la ejecución de JavaScript

3.1.2. Interfaces hardware

La aplicación no requiere hardware especializado más allá de un servidor web y una base de datos para almacenar la información de las partidas, usuarios y objetos del juego.

3.1.3. Interfaces software

RQ4. Deberá incluir soporte para APIs que permitan a aplicaciones externas acceder a todas las funcionalidades de la aplicación. Lo único que no permitirá será el acceso y participación en partidas.

3.1.4. Interfaces de comunicaciones

RQ5. La aplicación ofrecerá un servicio web basado en REST para permitir la obtención de datos del usuario y poder reportar problemas.

3.2 Características del sistema

La organización de los requisitos funcionales del sistema se basa en los principales escenarios de utilización:

- Inicio y cierre del sistema.
- Gestión de usuarios y administradores
- Administración de partidas y plantillas.
- Interacción con el tablero y los elementos de una partida

3.2.1 Inicio y cierre del sistema

ESC1. Inicio del sistema

- Propósito: Configurar el entorno necesario para que el sistema esté operativo.
- RQs asociados:

- **RQ6.** El sistema deberá comprobar la validez de los ficheros de configuración al iniciar.
- **RQ7.** Se debe crear un usuario administrador inicial si no existe.
- **RQ8.** Se debe crear una plantilla de partida, que será la plantilla de una partida por defecto

ESC2. Cierre del sistema

- Propósito: Asegurar el guardado y cierre adecuado de las sesiones activas, chats y partidas jugadas.
- RQs asociados:
 - **RQ9.** Al cerrar el sistema, se deberán guardar correctamente todos la información de las cuentas, partidas y plantillas del juego.

3.2.2 Gestión de cuentas

ESC3. Funciones de los usuarios y administradores

- RQs asociados a todas las cuentas:
 - **RQ10.** Todas las cuentas podrán registrarse, iniciar sesión y cerrar sesión.
 - **RQ11.** Todas las cuentas podrán cambiar su contraseña.
- RQs asociados a las cuentas de tipo usuario:
 - **RQ12.** Cada usuario tendrá una lista de amigos que son otros usuarios, mediante la cual podrá enviar solicitudes de amistad a usuarios, que podrán ser aceptadas o rechazadas.
 - **RQ13.** Los usuarios podrán crear sus propias plantillas, que serán utilizadas para la creación de sus partidas privadas.
 - **RQ14.** Los usuarios tendrán un nivel, que será la suma del puntaje que ha obtenido en partidas publicas anteriores.
 - **RQ15.** Los usuarios podrán interactuar con un sistema de reporte de problemas, mediante el cual enviaran el problema que han tenido introduciendo una descripción, y se les asignara un administrador por vía chat para poder comunicarse.
- RQs asociados a las cuentas de tipo administrador:
 - **RQ16.** Los administradores podrán eliminar cualquier cuenta de tipo usuario de la aplicación si así lo consideran oportuno.

- **RQ17.** Se les será asignado los problemas que tengan las cuentas de tipo usuario en la aplicación, tratando por vía chat dicho problema con el usuario afectado.
- **RQ18.** Podrán unirse tantos administradores como sea, a partidas públicas que se estén jugando actualmente para supervisarlas.
- **RQ19.** Un administrador no puede unirse a una partida privada.
- **RQ20.** Podrán consultar todos los datos de las partidas públicas ya finalizadas.

3.2.3 Administración de partidas

ESC5. Creación y gestión de partidas

- RQs asociados a todas las cuentas:
 - **RQ21.** Las partidas pueden ser públicas o privadas.
- RQs asociados a las cuentas de tipo administrador:
 - **RQ22.** Las partidas publicas estarán basadas en plantillas creadas por los administradores, por lo que los administradores pueden tanto crear como eliminar plantillas.
 - **RQ23.** Los administradores no podrán eliminar la plantilla por defecto.
 - **RQ24.** Un administrador podrá unirse a una partida en cualquier momento del transcurso de esta, desde que entra el primer jugador, hasta que termina la partida.
- RQs asociados a las cuentas de tipo usuario:
 - **RQ25.** Las cuentas de tipo usuario, podrán tanto crear como unirse a partidas privadas.
 - **RQ26.** Un usuario podrá invitar a otros usuarios a su partida privada que acaba de crear mediante la generación de un código (identificador único) de la partida privada, que el resto de los usuarios pueden utilizar para unirse, o enviando una invitación a la partida a algún usuario de su lista de amigos.
 - **RQ27.** Los usuarios podrán buscar partida publica, y les será asignada una partida en la que la media de nivel medio de los usuarios en dicha partida sea el más parecido posible al nivel del usuario que busca partida, siendo como mucho la diferencia desde 2 niveles menos al nivel del usuario hasta 2 niveles más, priorizando las partidas en las que la diferencia de nivel sea menor.

- **RQ28.** Los usuarios podrán elegir que plantilla utiliza la partida que quieren buscar, de entre todas las plantillas disponibles.

3.2.4 Interacción con el tablero y elementos de la partida

ESC7. Interacción con el chat

- Propósito: Permitir a los jugadores comunicarse entre sí vía mensajes de texto.
 - **RQ29.** Cada partida tendrá un chat, en el que podrán tanto enviar como leer mensajes de este, usuarios y administradores que se encuentren en dicha partida.

ESC7. Inicialización de la partida

- Propósito: Poner en igualdad de condiciones a todos los jugadores (cuentas de tipo usuario) de la partida.
- RQs asociados a las cuentas de tipo usuario:
 - **RQ30.** Al comenzar la partida, se repartirán las cartas que define la plantilla de esa partida a los jugadores equitativamente, además de asignar el tablero de la plantilla a la partida, y se mostrarán las cartas sobrantes.
 - **RQ31.** Las partidas serán de 2 a 8 jugadores, y al llegar a 2 jugadores se esperará 1 minuto hasta que se unan más jugadores para empezar la partida.
 - **RQ31.** Cada usuario comenzara en la casilla de salida.
 - **RQ32.** El turno de cada jugador se elegirá al azar.
 - **RQ33.** Se asociará a cada jugador un rol en la partida, que coincidirá con los personajes en las cartas de tipo personaje.

ESC7. Movimiento, acción y acusaciones en el tablero

- Propósito: Permitir a los jugadores interactuar con el tablero de juego por turnos.
- RQs asociados a las cuentas de tipo usuario:
 - **RQ34.** Cada usuario podrá saltar su turno indicándolo, sin surtir ninguna acción en ese turno.
 - **RQ35.** Los usuarios deberán respetar un turno de juego. Se dispondrá de 1 minuto para hacer dicho turno, si se pasa el minuto la acción contará como pasar turno, y si esto sucede 3 veces, el usuario será expulsado de la partida automáticamente.

- **RQ36.** El usuario podrá hacer la acusación final en cualquier momento de su turno, siempre y cuando no haya hecho él una acusación previamente en ese mismo turno.
- **RQ37.** Cada usuario podrá primeramente lanzar los dados, y el valor obtenido será la distancia máxima que podrá recorrer en el tablero, pudiendo recorrer menos si así lo desea.
- **RQ38.** Para moverse a una habitación hace falta llegar a la puerta de dicha habitación, y moverse una casilla más para dentro de la habitación (por lo que hará falta para entrar en una habitación la distancia desde la posición del jugador hasta la puerta + 1 como mínimo).
- **RQ39.** El usuario podrá hacer una acusación normal (no final) si se encuentra en una habitación. Esa acusación tendrá que llevar una carta con la habitación en la que se encuentra.
- **RQ40.** Tras una acusación, se deberá permitir al resto de jugadores mostrar una carta que coincida con la acusación si tienen una que coincida, si tienen 2 que coincidan mostrar la que elijan pudiéndose repetir de anteriores veces (si anteriormente la ha desvelado, puede desvelarla otra vez en este contexto), o si no tienen ninguna coincidente, no mostrar ninguna.
- **RQ41.** En el caso de que un usuario exceda un tiempo de 30 segundos para desvelar una carta (si es que debe), se mostrara automáticamente la carta coincidente en el caso de que haya únicamente una, y una al azar en el caso de que haya varias.
- **RQ42.** Todas las cartas desveladas podrán ser leídas únicamente por el jugador que ha hecho la acusación previa, y en su defecto por el administrador/es que haya en la partida.
- **RQ43.** Si una acusación final es verdadera, entonces el usuario gana la partida, si no la perderá y será expulsado de la misma. Dicha acusación no podrá ser vista por ningún usuario a no ser que esta acusación provoque la victoria de este.
- **RQ44.** Tras realizar una acusación de cualquier tipo, el usuario pasara el turno al siguiente jugador.
- **RQ45.** Cada acción en la partida que hace cada usuario tendrá una puntuación :
 - **Ganar la partida:** 200 puntos
 - **Cada carta desvelada:** 10 puntos

ESC8. Consulta de datos durante la partida

- Propósito: Permitir a las cuentas de usuario y de administrador consultar datos sobre la partida, sin tener por qué ser su turno en el caso de los usuarios.
- RQs asociados a cuentas de usuario:
 - **RQ46.** Un usuario podrá consultar una lista de cartas descartadas como posibles implicadas en los asesinatos, dicha lista se actualizará conforme vaya desvelando cartas y descartando las mismas. Estas le serán de ayuda para ir descubriendo el verdadero asesinato.
 - **RQ47.** Cada usuario podrá ver la acusación actual que se está haciendo para saber que cartas puede elegir para desvelar.
 - **RQ48.** En cualquier momento, un usuario podrá comprobar en que posición se encuentran el resto de los jugadores en la partida, su rol, si están jugando, su puntuación, si han sido expulsados y la razón por la que han sido expulsados.
- RQs asociados a las cuentas de tipo administrador:
 - **RQ49.** Un administrador podrá expulsar a un jugador en el transcurso de la partida si así lo considera, debiendo exponer la razón de expulsión.
 - **RQ50.** El administrador podrá obtener todos los datos de la partida, incluidos los datos que pueden consultar todos los usuarios, y también las últimas cartas que han sido desveladas y las cartas de todos los usuarios.

ESC7. Finalización de una partida

- Propósito: definir las acciones resultantes tras la finalización de una partida.
- RQs asociados a las cuentas de tipo usuario:
 - **RQ51.** Cuando gane un jugador, se acabará la partida.
 - **RQ52.** Al finalizar una partida publica, se hará un recuento de todos los puntos obtenidos por los jugadores, y se le sumaran al nivel del jugador, de tal manera que cada 500 puntos, se subirá al nivel siguiente el jugador.
 - **RQ53.** En las partidas privadas, los usuarios no obtienen puntos.
 - **RQ54.** Si un jugador es expulsado, recibirá -50 puntos.

3.3 Requisitos de prestaciones

- **RQ55.** El sistema debe ser capaz de manejar un mínimo de 50 partidas simultáneas con un promedio de 6 jugadores por partida.

- **RQ56.** Las respuestas del servidor a las acciones de los jugadores no deberán exceder 2 segundos en condiciones normales.

3.4 Requisitos de persistencia

- **RQ57.** Los datos de cuentas, partidas y plantillas serán almacenados en una base de datos relacional.
- **RQ58.** Los logs de las partidas deberán ser almacenados durante un período mínimo de 30 días.

3.5 Requisitos de diseño

- **RQ59.** La aplicación deberá desarrollarse utilizando tecnologías multiplataforma.
- **RQ60.** La arquitectura deberá seguir el patrón MVC (Modelo-Vista-Controlador).
- **RQ61.** Deberá garantizarse la separación entre lógica de negocio y lógica de presentación.

3.6 Atributos del sistema

3.6.1 Fiabilidad

- **RQ62.** La aplicación deberá garantizar un uptime del 99,5%.

3.6.2 Disponibilidad

- **RQ63.** El sistema deberá ser accesible 24/7 desde cualquier dispositivo con conexión a internet.
- **RQ64.** En caso de caída del sistema, deberá recuperarse automáticamente en un tiempo máximo de 10 minutos.

3.6.3 Seguridad

- **RQ65.** Todas las contraseñas deberán almacenarse cifradas.
- **RQ66.** Las sesiones de usuario deberán expirar tras 1 hora de inactividad.
- **RQ67.** Deberá implementarse protección contra ataques de inyección SQL y XSS.

3.6.4 Mantenibilidad

- **RQ68.** El código deberá seguir estándares de programación y estilo para facilitar la comprensión y el mantenimiento futuro.

3.6.5 Escalabilidad

- **RQ69.** La arquitectura deberá permitir la adición de nuevos servidores para manejar el crecimiento del tráfico sin interrupciones.

3.6.6 Portabilidad

- **RQ70.** El sistema deberá poder migrarse a una infraestructura en la nube sin necesidad de modificaciones significativas en el código base.

4. DISEÑO

4.1 Diagramas de casos de uso

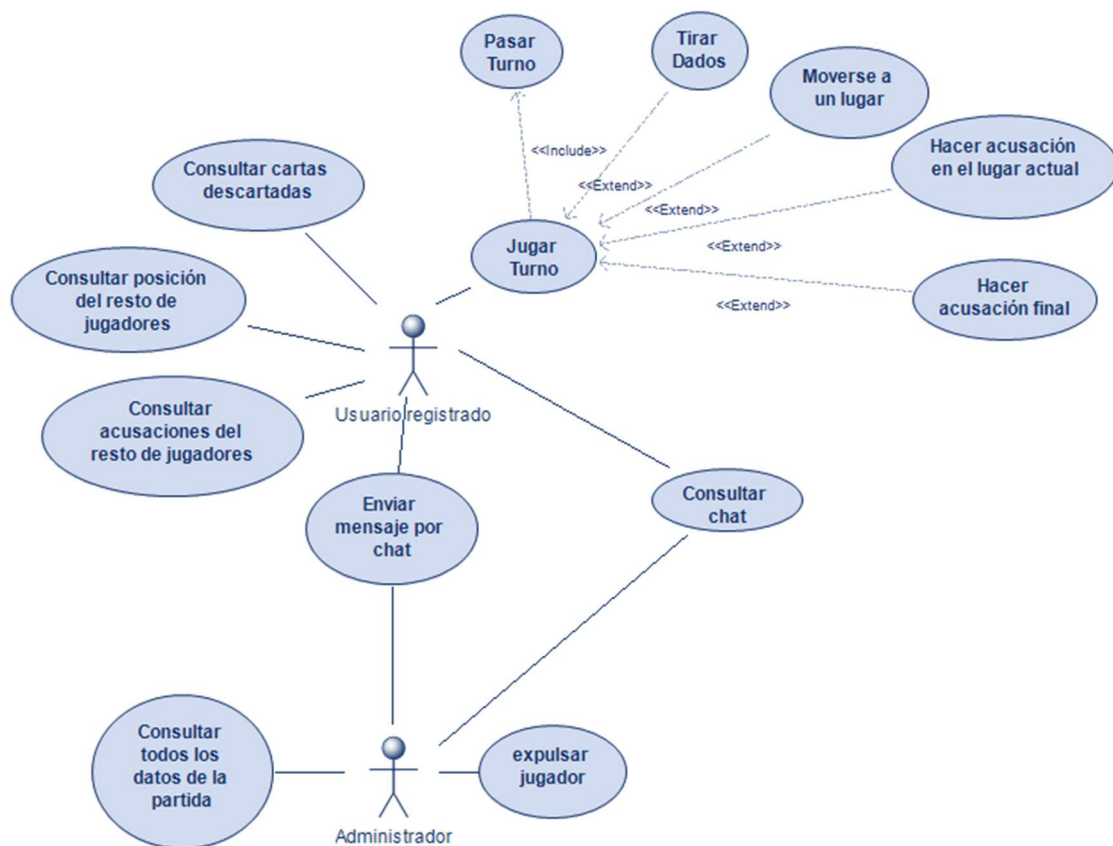


Ilustración 2: diagrama de caso de uso para el desarrollo durante una partida

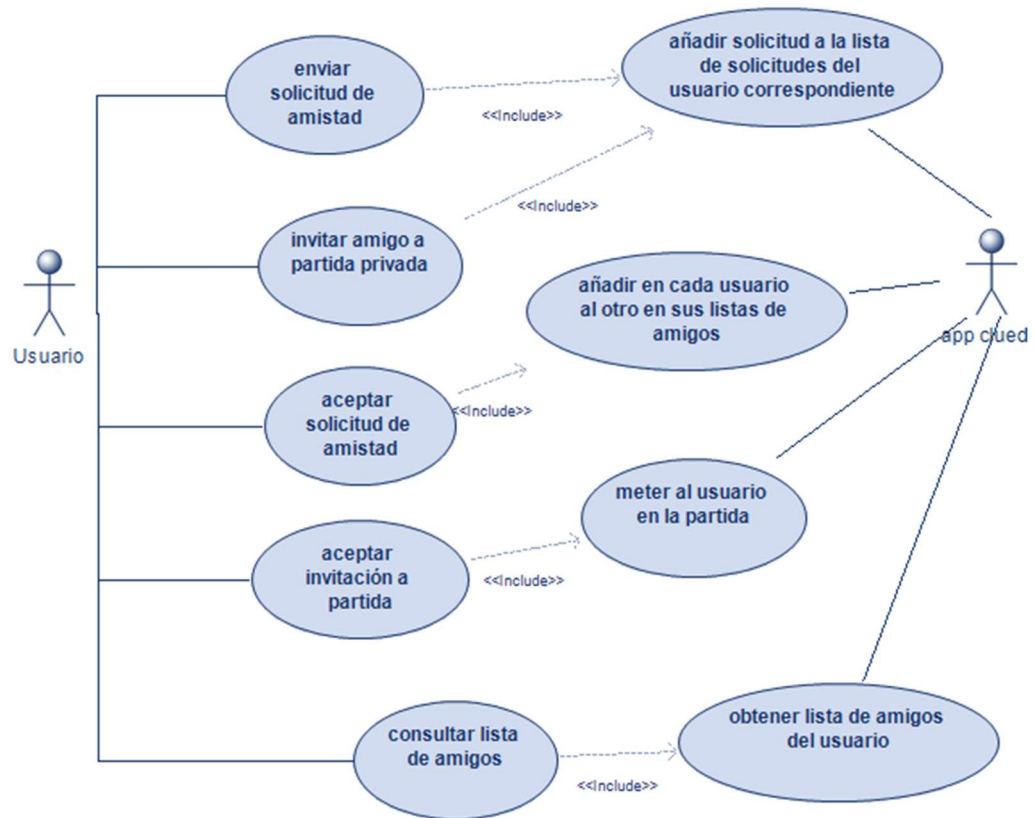


Ilustración 3: diagrama de caso de uso para la gestión de la lista de amigos

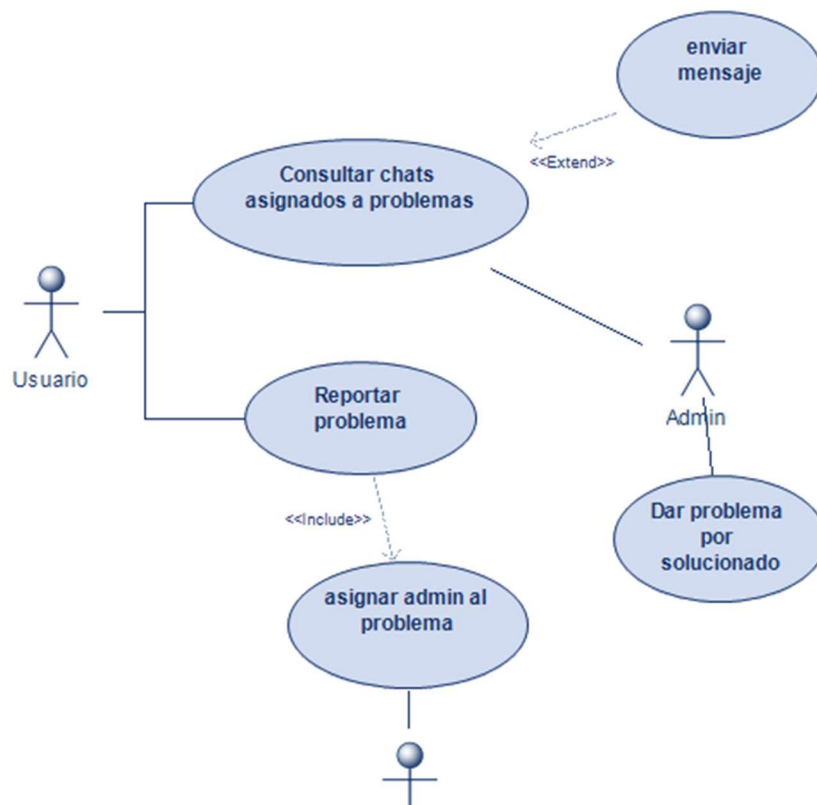


Ilustración 4: diagrama de caso de uso para el reporte de problemas

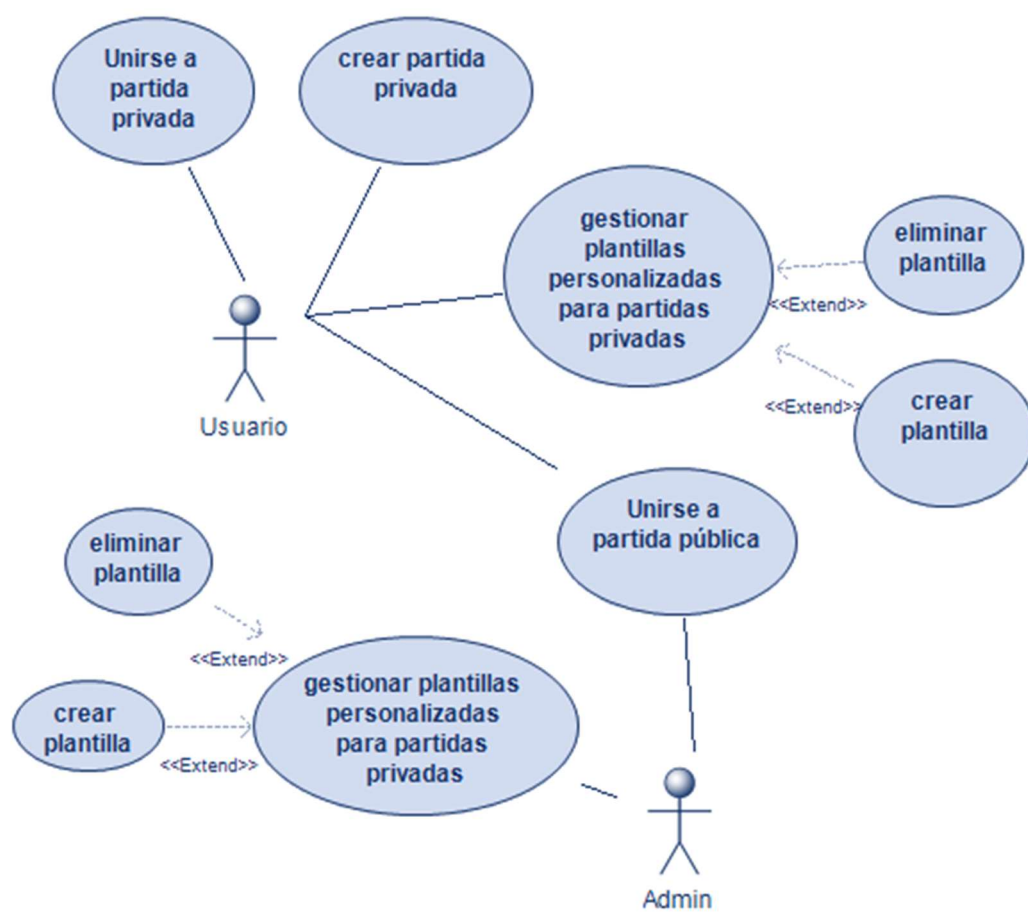


Ilustración 5: diagrama de caso de uso para la gestión de partidas

4.2 Diagrama de clases

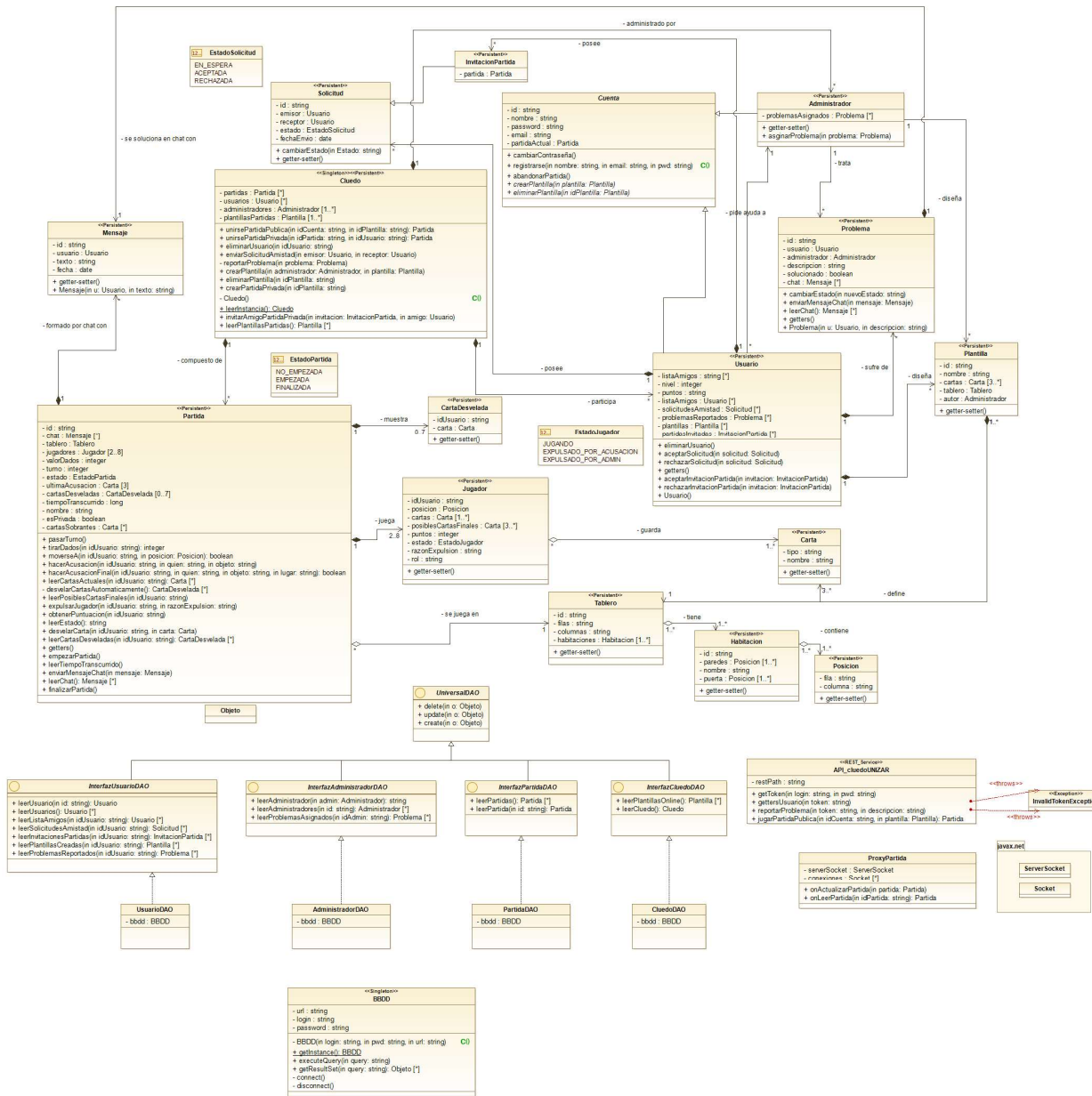


Ilustración 6: diagrama de clases de la aplicación

4.3 Diagramas de actividades

Los tres casos de uso que se han considerado importantes explicar con diagramas UML han sido el inicio de una partida publica, la interacción con una partida, y el reporte de un problema presente en un usuario.

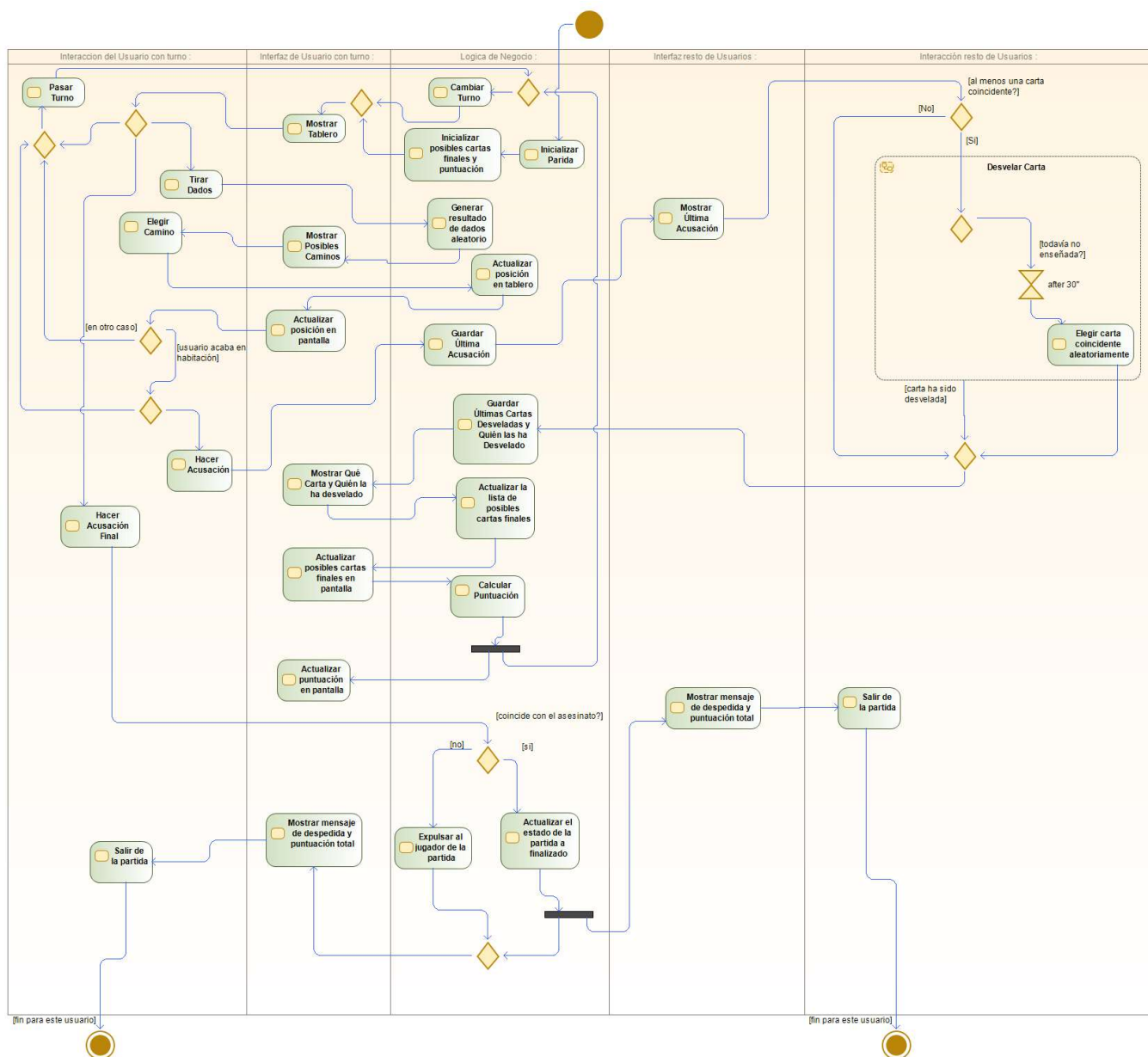


Ilustración 7: diagrama de actividades para el desarrollo durante una partida

Considerando no hacer más complejo el diagrama anterior, se ha planteado abstraer la funcionalidad de pasar turno tras 1 minuto, y la acumulación de estas acciones para la expulsión de dicho jugador en otro diagrama. “turno completado?” se refiere a un turno en el que el jugador a pasado el turno, o ha realizado una acusación de cualquier tipo. En cualquier otro caso, se considerará que no se ha completado el turno.

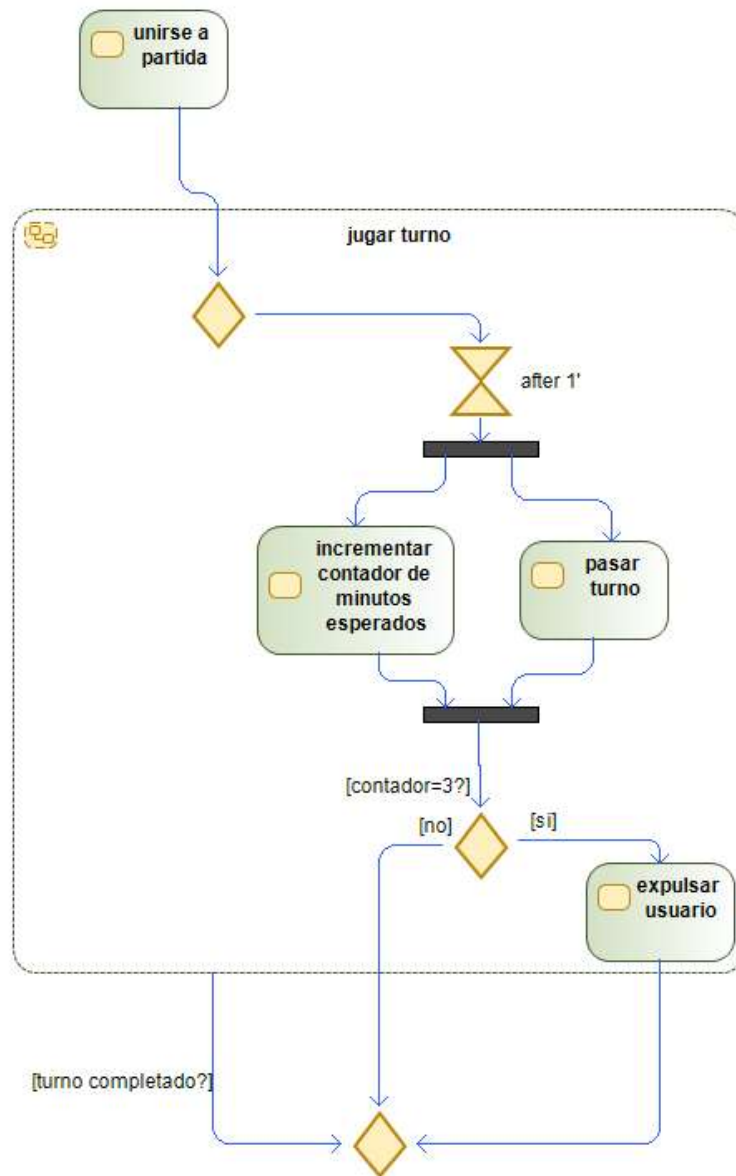


Ilustración 8: diagrama de actividades específico para el tiempo de espera por turno

El siguiente caso es para un usuario que reporta un problema, e interactúa con el chat directamente, no se ha incluido en el diagrama, pero podría dejar el chat abierto e ir a hacer otra cosa en la aplicación, pudiendo acceder al chat cuando se desee por los usuarios y administradores.

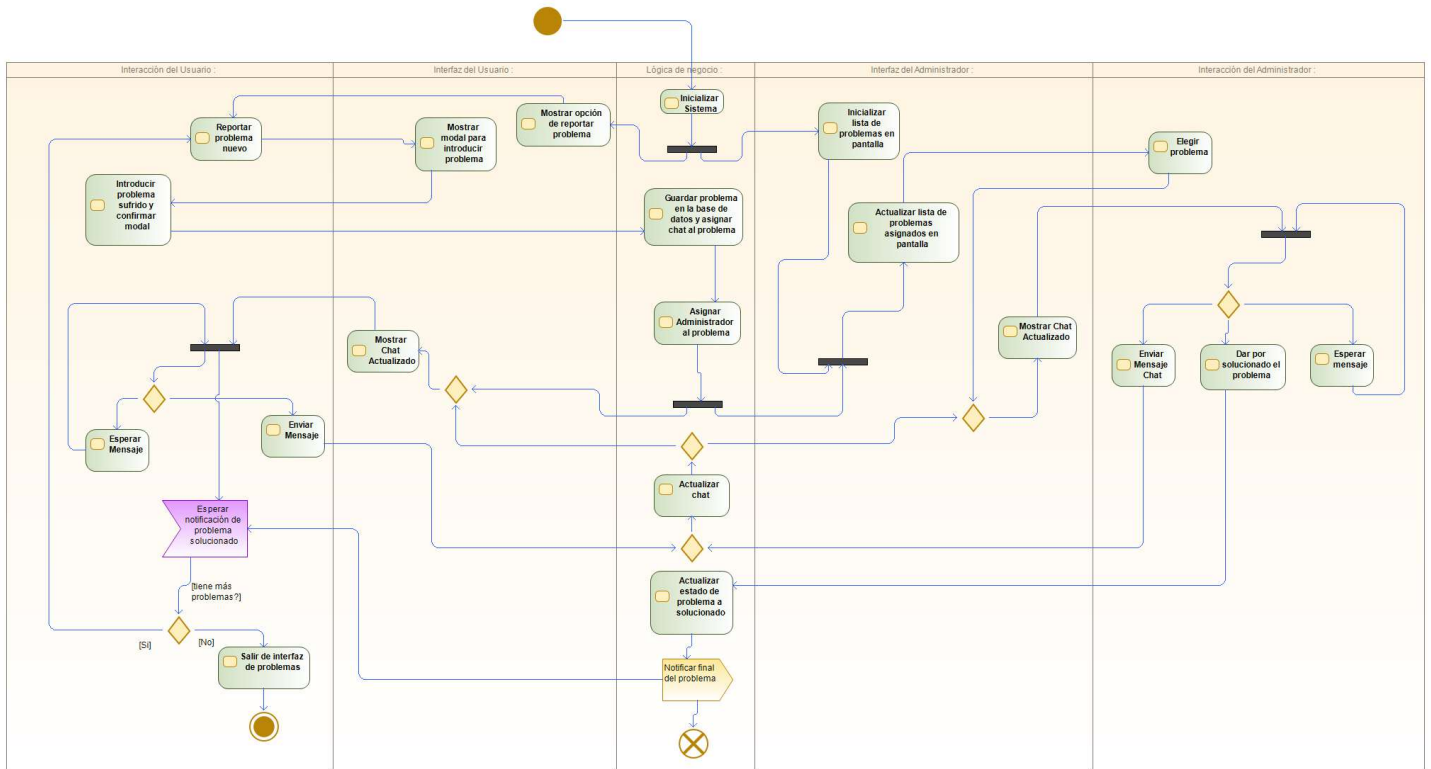


Ilustración 9: diagrama de actividades para el reporte de problemas de usuarios

El tercer caso es para la inicialización de una partida del juego:

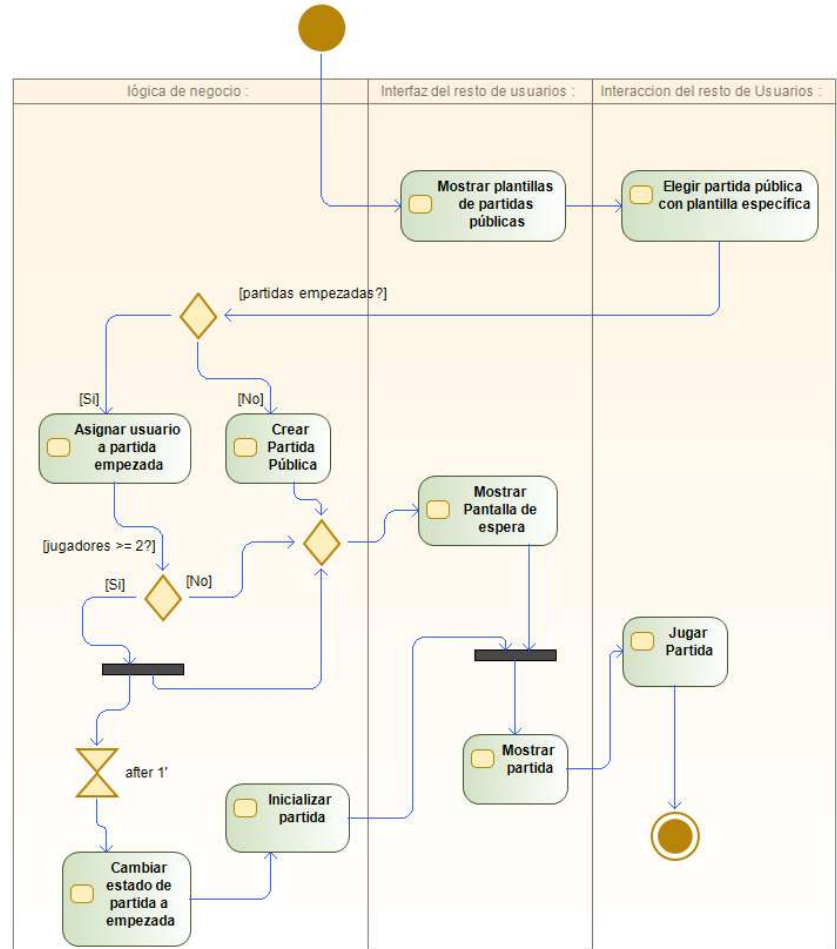


Ilustración 10: diagrama de actividades para la inicialización de una partida en juego

4.4 Diagramas de secuencia

Los diagramas siguientes representan la lógica de negocio para cada caso de uso elegido.

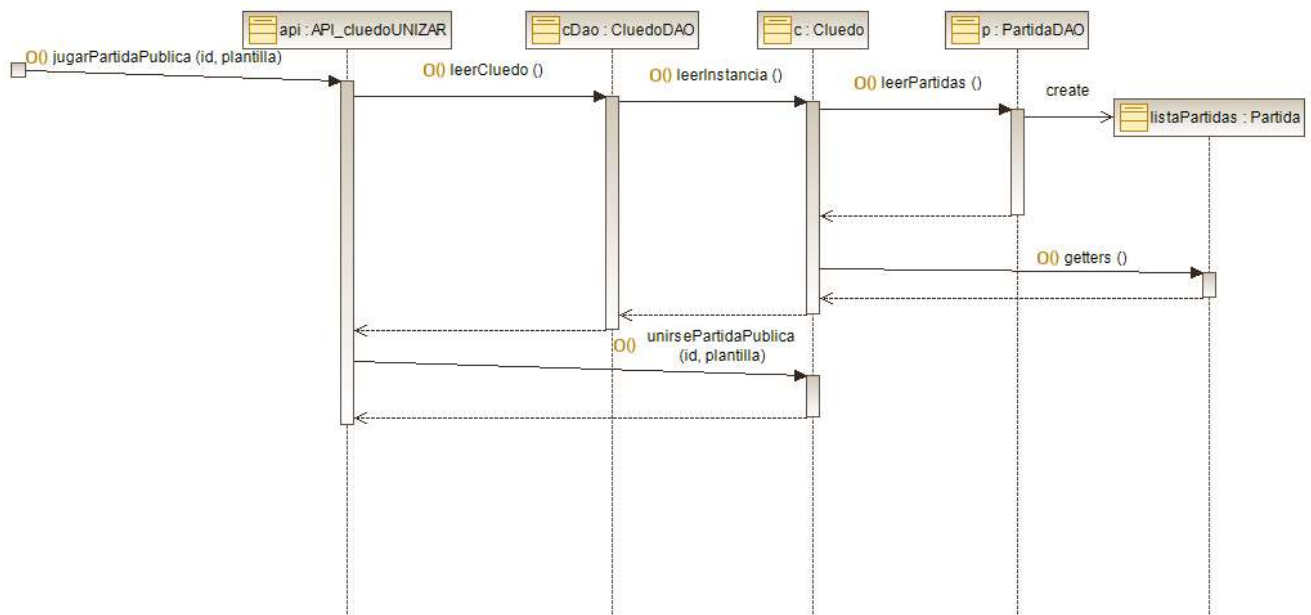


Ilustración 12: diagrama de secuencia para la inicialización de una partida publica

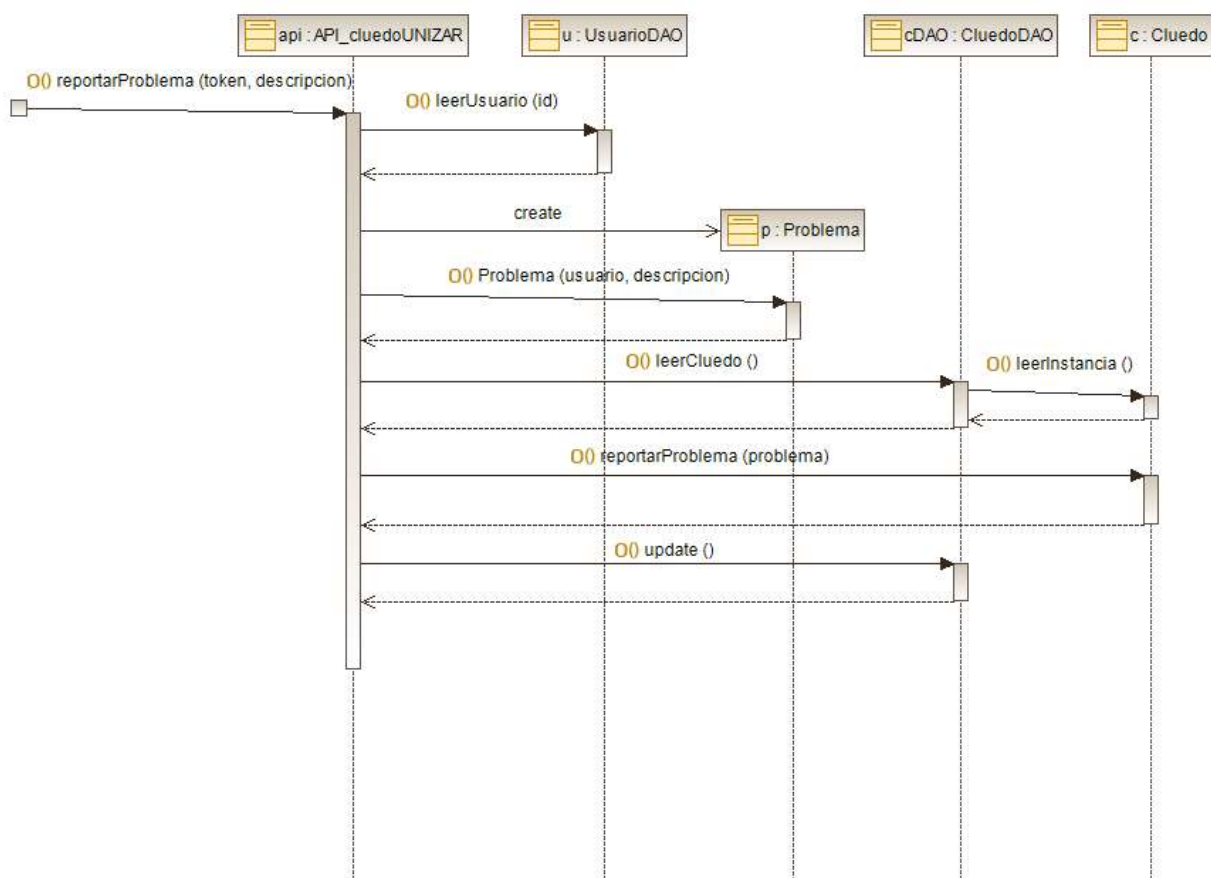


Ilustración 11: diagrama de secuencia para el reporte de un problema

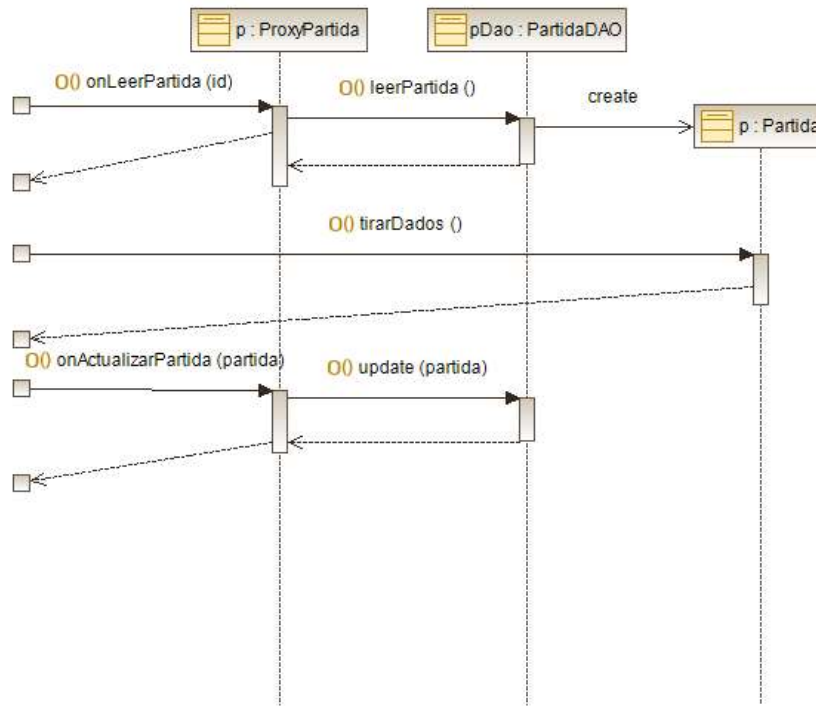


Ilustración 13: diagrama de secuencia para la interacción en una partida

Para este diagrama se ha considerado unicamente la accion de tirar dados y de obtener la instancia de la partida actualizada. Cada vez que el usuario quiera hacer una accion concreta, debera leer la instancia de la partida en ese momento, interactuar con la partida mediante sus metodos, y actualizarla en la base de datos con actualizar partida, por lo que en un turno si se quiere pasar el turno, enviar un mensaje al chat, hacer una acusacion o desvelar cartas, el procedimiento seria el mismo que el mostrado para tirar los dados.