

A dark blue vertical bar runs down the left side of the page. A blue arrow points to the right from this bar, containing the date.

27-3-2017

Texas Hold'em Poker

CPIFP Los Enlaces

Several thin, curved lines in dark blue and light grey originate from the bottom left and sweep upwards and to the right.

Mario Codes Sánchez

DESARROLLO DE APLICACIONES MULTIPLATAFORMA

Contenido

Documento descripción del proyecto	2
Contexto del Proyecto	2
Ámbito y Entorno	2
Análisis de la realidad	2
Solución y justificación de la solución propuesta	2
Destinatarios	2
Objetivo del proyecto.....	3
Documento de acuerdo del proyecto	4
Requisitos funcionales y no funcionales	4
Tareas	4
Metodología a seguir para la realización del proyecto	4
Planificación temporal de tareas.....	4
Presupuesto	5
Contrato y pliego de condiciones.....	5
Análisis de riesgos	5
Documento de análisis y diseño.....	6
Análisis y diseño de la arquitectura de la aplicación.....	6
Tecnologías y herramientas usadas y descripción de las mismas.....	6
Arquitectura de componentes de la aplicación	6
Modelado de datos	7
Registro de un usuario nuevo	8
Sistema de inicio de sesión	9
Creación de partida nueva	10
Unirse a una partida ya creada	11
Thread independiente de estatus	12
Funcionamiento de ronda	13
Análisis y diseño del sistema funcional	8
Análisis y diseño de la interfaz de usuario	14
Menú Principal	14
Ventana de Juego	15

Documento descripción del proyecto

Contexto del Proyecto

Ámbito y Entorno

Este proyecto se encuentra dentro del ámbito de juegos multijugador en línea, en un entorno variable según se desee. Aunque se ha diseñado para poder ejecutarse, dado el caso, a nivel global previa configuración y puesta a punto de servidores, los cuales dispongan de potencia y/o requerimientos suficientes para hacerse cargo del nivel de flujo de peticiones requeridas; También se puede usar a nivel más personal, dentro de una propia red local sin necesidad de disponer de acceso a internet.

Análisis de la realidad

Es un proyecto que por la forma en la que está diseñado, puede ser fácilmente llevado a otros sistemas más portables, como por ejemplo Android. Que quizás disponen de un abanico más amplio en el mercado, que el existente actual para las aplicaciones en Java como tal. Dicho esto, él proyecto también se encarga de cubrir un pequeño hueco, como juego de cartas disponible en línea para Java Runtime.

Actualmente que se hayan realizado en Java, no hay ninguna aplicación o juego de aplicaciones famoso, que se encargue de cubrir este hueco en el mercado. Ya que competencia como tal no hay, es una carencia que se cubre con el desarrollo y lanzamiento de este Póker.

Solución y justificación de la solución propuesta

Se han cubierto un poco todos los frentes necesarios hoy en día en cuanto a tecnologías, para disponer de además de una aplicación robusta, presencia en internet mediante la página web. De esta manera es más fácil llamar la atención del usuario objetivo, y se puede publicitar la aplicación y hacerla visible de una manera más eficaz.

Destinatarios

Público el cual disfrute, tanto con los juegos de cartas, como con una buena partida en línea ya sea entre sus amigos y conocidos o con terceras personas.

Esta aplicación ha sido diseñada con ámbito global. El idioma en el que está escrito el código es inglés (recordar que es open-source) y el idioma de la interfaz es el inglés también. Más adelante tengo la intención de traducirla al español y ofrecer la opción de cambiarlo al ejecutarse la app.

Objetivo del proyecto

Mi objetivo mediante este proyecto es demostrar la efectividad y capacidad de desarrollo que se puede conseguir mediante los lenguajes de programación aprendidos a lo largo del correspondiente curso de Desarrollo de Aplicaciones Multiplataforma. Más en concreto utilizando Java como lenguaje base, así como diferentes herramientas de desarrollo adicionales y también la implementación de diferentes patrones de programación y diseño que sirvan para reducir en la medida de lo posible la complejidad lógica del proyecto.

En cuanto a la orientación práctica se desea desarrollar e implementar un sistema con estructura Cliente – Servidor y en el cuál el Cliente se diseñe siguiendo el Modelo Vista – Controlador. Toda la lógica del juego, el cambio del mismo por sus diferentes fases, así como procesamiento en tiempo real de las acciones de cada jugador será realizado por el Servidor. Dedicando de esta forma el cliente de forma casi exclusiva a realizar conexiones con el servidor para enviar las acciones del jugador y recibir las respuestas de éste.

También se desea tener en alta estima la seguridad de la información sensible introducida por los potenciales usuarios, por lo cual se aplicarán técnicas de procesamiento y almacenamiento seguras de todo lo relacionado a usuarios, contraseñas etc.

Por último, adquirir y afianzar conocimientos en lo relacionado al mundo de desarrollo de aplicaciones web; Ya que es una parte en la que ahora mismo antes de empezar el proyecto me encuentro bastante carente y me sirve como excusa para verme obligado a investigar tecnologías y lenguajes.

What I want to achieve, it's to show what can be done with the technologies I have learned during this whole two years. Those are mainly Java as programming language with several other tools such as programming patterns used to relieve a bit the logic aspect of this Project.

The system has been designed to have a Client – Server structure and following the Model View – Controller. All the heavy processing tasks are done by the Server also as the Flow of a game through all it's several phases. The client only has to send the actions the player has made to the server and take back the answer to know what to do.

Another objective is to learn and implement a safe way to save all the consumer's sensible data.

To end, I want to obtain knowledge about the web Development world, how's it done, technologies, frameworks and all that may be needed to implement it correctly.

Documento de acuerdo del proyecto

Requisitos funcionales y no funcionales

1. Establecer un sistema de usuarios para que cada persona disponga de su propia cuenta personal y se le diferencie.
2. Establecer forma segura de guardar las contraseñas en la base de datos.
3. Integrar el registro y creación de estas cuentas mediante la página web.
4. Integrar en la aplicación Java el sistema de partida por salas para que se puedan realizar diferentes partidas de forma simultánea sin que interfieran entre ellas.
5. Realizar un sistema de abandono para que un jugador pueda salir en cualquier momento sin que afecte a los demás en la sala.
6. Establecer en la página web la documentación e información necesaria del juego, así como distribuir el código de forma libre y gratuita.
7. Permitir realizar una partida de póker de forma correcta discerniendo claramente las condiciones de victoria y derrota, así como por las cuales el juego terminará en su totalidad.

Tareas

1. Planificación, documentación y segmentación de todas las tareas y funciones de las que se quiere que disponga el proyecto.
2. Realización del diagrama de clases de la parte en Java del proyecto.
3. Realización de los diagramas de secuencia de todas y cada una de las acciones posibles para el usuario.
4. Planificación y estructuración de todos los paquetes a enviar por red entre servidor y cliente, especificando en cada momento, el orden de datagramas a seguir.
5. Investigación e implementación de forma actual segura de guardar contraseñas.
6. Planificación y modelado de la base de datos.
7. Diseño y creación de la página web.

Metodología a seguir para la realización del proyecto

Metodología en cascada. El proyecto se divide en varias fases (requerimientos, diseño, implementación, verificación y mantenimiento) que se van a seguir y completar antes de comenzar con la siguiente.

Planificación temporal de tareas

- Planificación previa: 5 horas.
- Diseño: 22 horas.
 - Creación y estructuración de los diagramas a seguir: 15 horas.
 - Diseño de la estructura de la página web: 3 horas.
 - Diseño de la estructura de la base de datos: 1 horas.
 - Diseño de GUI: 3 horas.
- Investigación: 25 horas.
 - Metodologías de seguridad: 5 horas.
 - Aprendizaje de lenguajes web necesarios: 20 horas.
- Implementación: 65 horas.
 - Parte relacionada a Java: 40 horas.
 - Implementación de la página web: 25 horas.
- Documentación 8 horas.
- Problemas adicionales no previstos: 20 horas.

Presupuesto

- Compra del dominio para la página web: 10€.
- Hosting del dominio: 10€.
- Logos, iconos y fondos para la aplicación: 20€
- Horas de trabajo – 120 horas x 6€ hora: 720€
- **Total: 760€.**

La monetización de la aplicación será realizada de forma **totalmente voluntaria** por parte de los usuarios. Esto es a través de donaciones completamente opcionales ya que, personalmente, no me agradan para nada las aplicaciones que incluyen publicidad de forma agresiva y/o intrusiva que no resultan más que una gran molestia para el usuario. Considerando como cualquiera de este tipo: publicidad de Google, anuncios tipo ad etc. Como mucho podría considerar el lanzamiento de una segunda app de pago, con la misma funcionalidad que la primera, pero que tenga cambios de apariencia **puramente cosméticos**. Esto descarta diferencias o barreras de funcionalidad, resultados, o experiencia en la ejecución principal de la aplicación.

Haciendo una estimación base ofreciendo esta segunda modalidad de la aplicación a 3€ y suponiendo donaciones de 1€; Para rentabilizar la aplicación en pongamos, el plazo de un año haría falta una media de:

- 19 compras de la aplicación & 10 donaciones de 1€ / mes.

Contrato y pliego de condiciones

Cuando todas las tareas anteriormente especificadas se vean satisfechas de la forma correcta y sin tener comportamientos de software inesperados.

Análisis de riesgos

Dispone de un bajo riesgo económico, debido a la pequeña totalidad de la cantidad invertida. El mayor riesgo que existe se encuentra relacionado al tiempo, debido a que aunque muchas de las tecnologías necesarias me son ya conocidas, hay otras (todas las relacionadas a la parte web) que aunque las voy aprendiendo a un ritmo regular en mi entorno de desarrollo de las FCT; Me son por el momento totalmente desconocidas, por lo que las estimaciones de tiempo previstas para ellas pueden verse comprometidas, y su número límite de horas, verse sobrepasado debido a la probable cantidad de problemas imprevistos existentes.

Dicho esto, para intentar solventar en la medida de lo posible este inconveniente, he dejado todo lo relacionado en cuanto a la página web para la parte final; Ya que, aunque es una parte necesaria para el cumplimiento del proyecto en su totalidad, no es crítica para el desarrollo del mismo. Pudiendo de esta forma dejarlo para un momento futuro en el tiempo, en el cual haya podido suplir un poco esta carencia de conocimiento, con el adquirido en el entorno de trabajo actual.

Documento de análisis y diseño

Análisis y diseño de la arquitectura de la aplicación

Tecnologías y herramientas usadas y descripción de las mismas

- Aplicación:
 - Java – Usado como lenguaje principal de programación.
 - IntelliJ IDEA – Entorno de desarrollo en Java.
 - PlantUML – Plugin completo para este IDE que permite la generación de todo tipo de diagramas.
- Base de Datos:
 - MySQL – Gestor de la base de datos, usado para la persistencia de datos.
- Página Web:
 - HTML5 – Forma la estructura de la página.
 - LESS – En conjunto a CSS3 formarán los diferentes Layouts.
 - Javascript – Validación de datos.
 - PHP – Programación relacionada al acceso de la base de datos.
 - [Posible] AJAX – Para hacer la página web más dinámica.
 - PhpStorm – IDE usado.

Arquitectura de componentes de la aplicación

Se usará una estructura cliente – servidor. Por un lado, se encuentra el servidor que almacena la página web, la base de datos y la forma de interactuar entre ambas, y por otra el cliente en Java para acceder a ambas. Dado el caso se pueden encontrar todos en la misma máquina, pero seguirán existiendo la base de esta arquitectura.

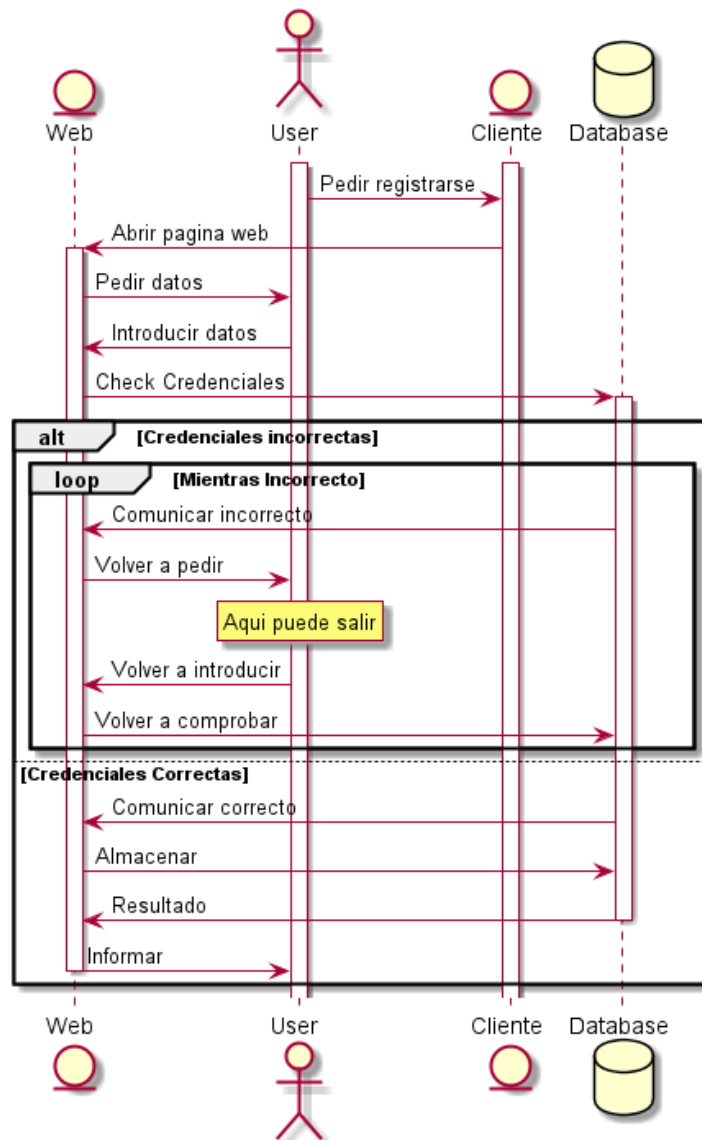
Modelado de datos

(Por hacer)

Análisis y diseño del sistema funcional

Registro de un usuario nuevo

Menu de Usuario - Registro



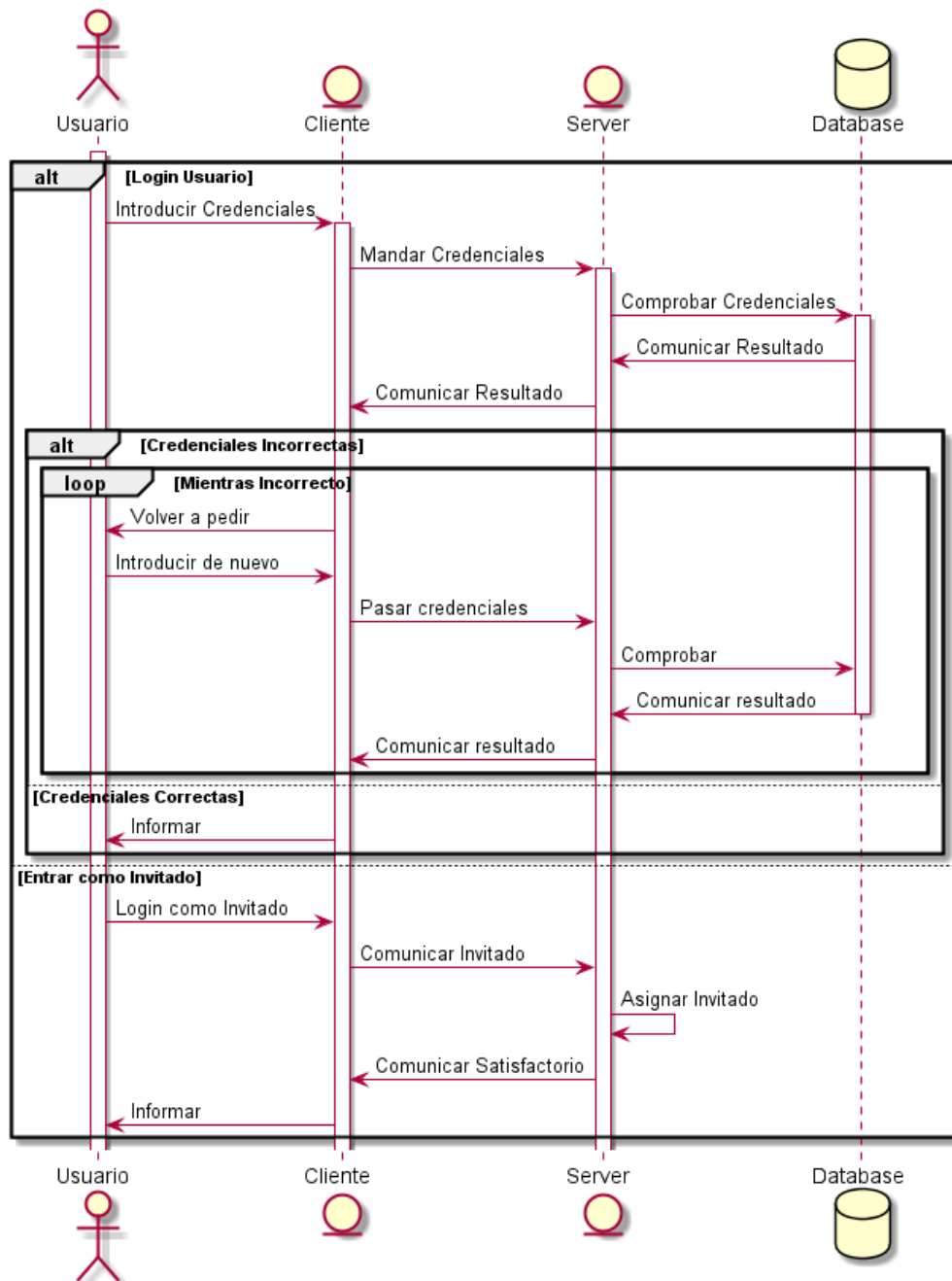
Se puede realizar desde la web directamente, o se dispone una opción para ello en el cliente de la aplicación, la cual abrirá la web cuando se seleccione.

Se introducen los datos en la web, esta hace la validación de los datos y cuando todo está correctamente rellenado, pasa a hacer la validación de usuario disponible contra la base de datos.

Las contraseñas nunca se almacenan en texto plano. Guardo un 'salted hash'. Esto es el hash de la contraseña, al cual se le ha añadido un prefijo aleatorio que se almacena en la cuenta de usuario. Cuando el usuario intenta entrar en la cuenta, se le aplica el prefijo, se saca el hash y se compara con el almacenado. De ésta forma si por cualquier error de seguridad se puede acceder o descargar mi base de datos, se evitan los sistemas de descifrado de contraseñas previamente 'hasheadas' (conocidos como 'Rainbow tables') y habría que aplicar sistemas de fuerza bruta de forma individual a todas y cada una de las contraseñas. Lo cual lo hace poco viable intentarlo.

Sistema de inicio de sesión

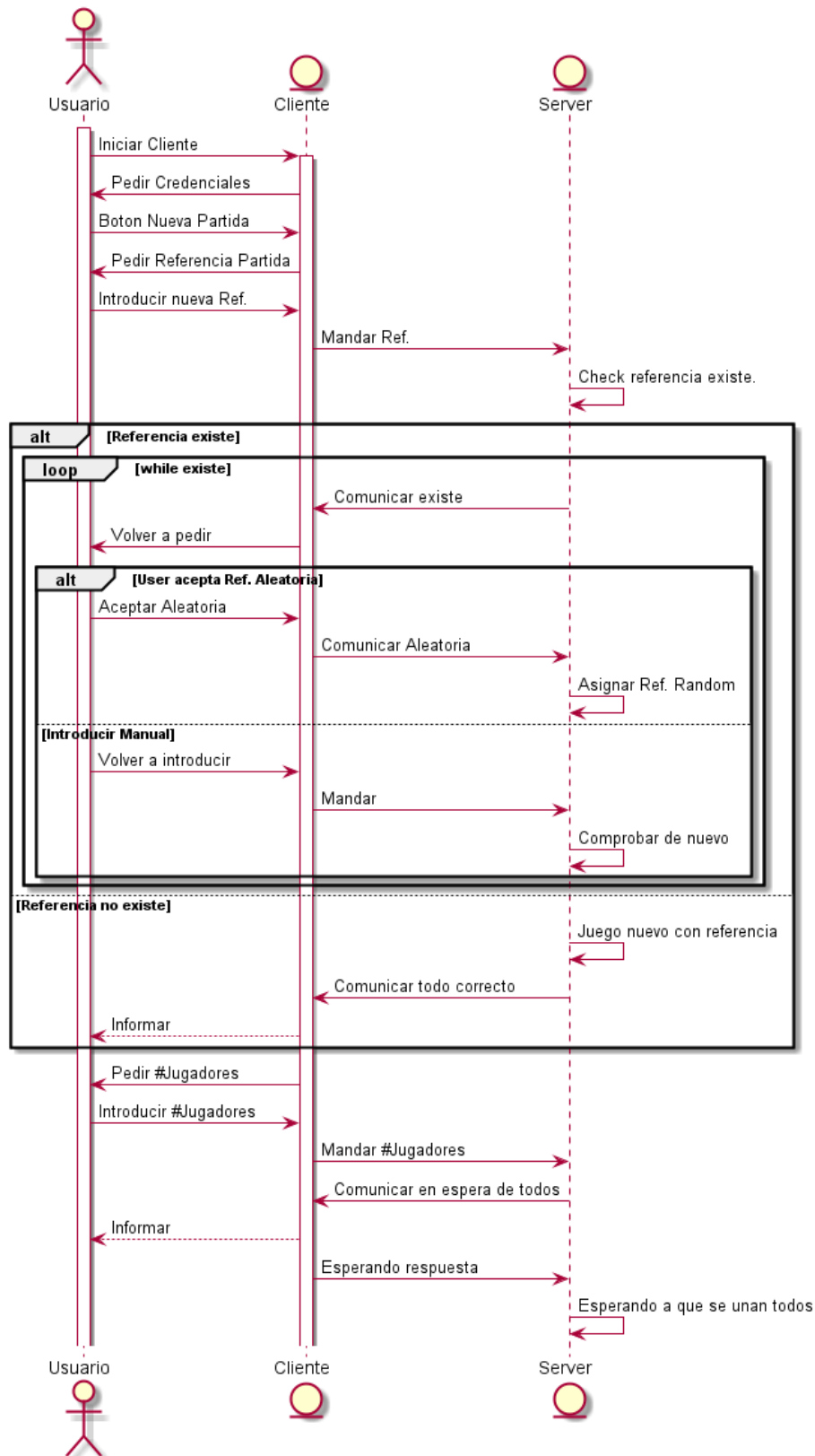
Parte común - Login



El usuario lo realiza desde la propia aplicación en Java; El cliente le pasa la información al servidor, quien la valida en la base de datos. Si decide que no quiere introducir sus datos puede jugar como usuario invitado; El cliente le comunica esto al servidor, el cual le asigna una ID temporal.

Creación de partida nueva

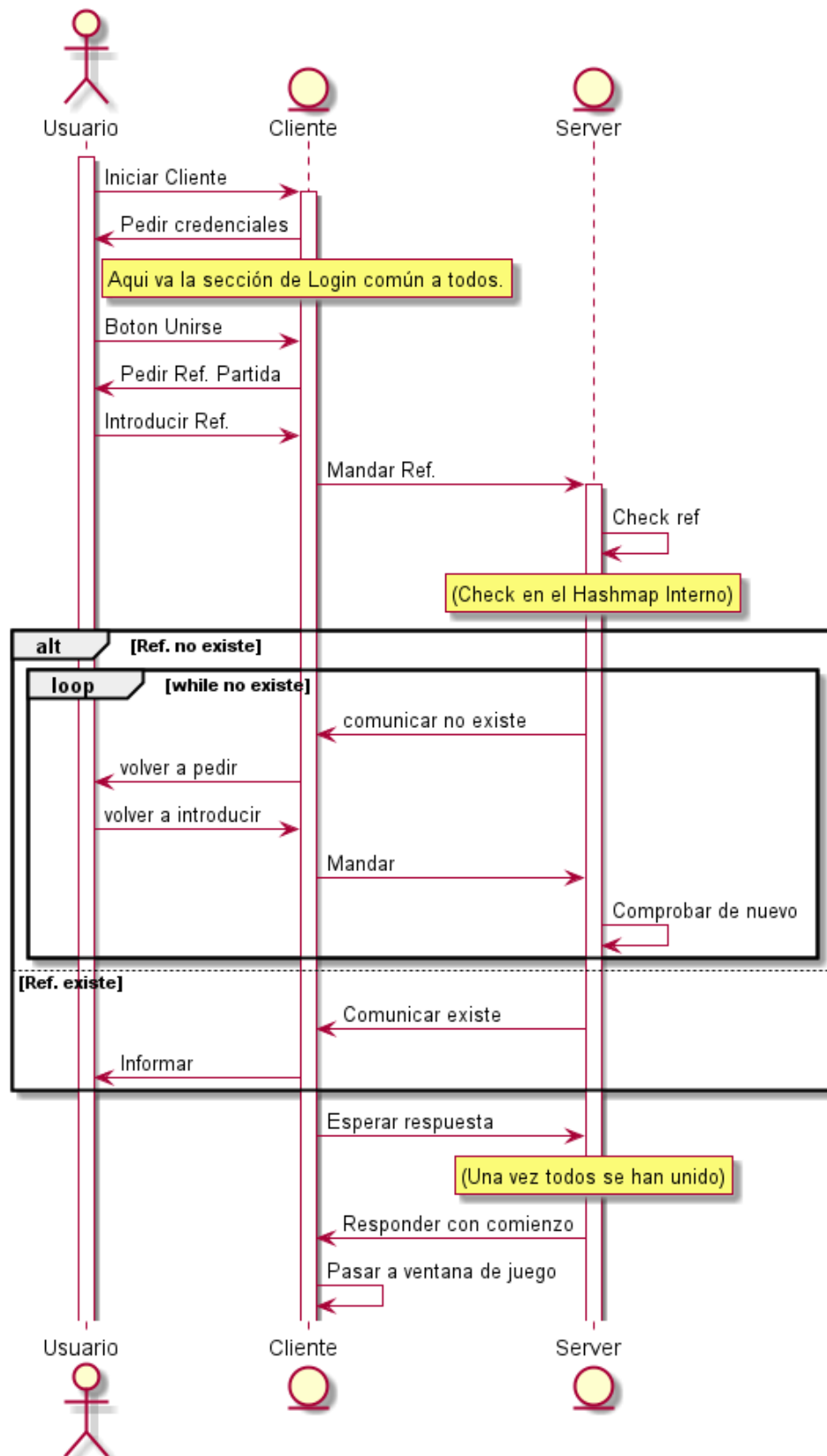
Menu Principal - Partida Nueva



Todos los usuarios registrados que quieran, pueden crear una partida nueva. Simplemente deben introducir los datos que les va pidiendo el servidor. La partida comenzará una vez se hayan unido todos los jugadores previstos.

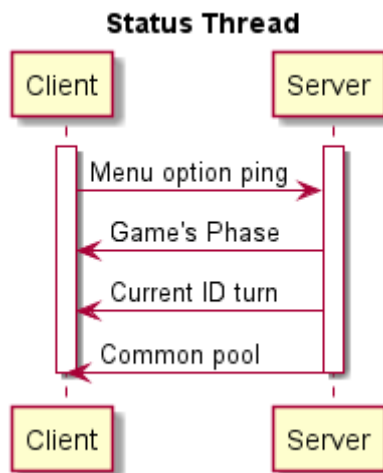
Unirse a una partida ya creada

Menu Principal - Unirse a Partida



Pasos para unir a un jugador a una partida que ha sido previamente creada por un jugador registrado.

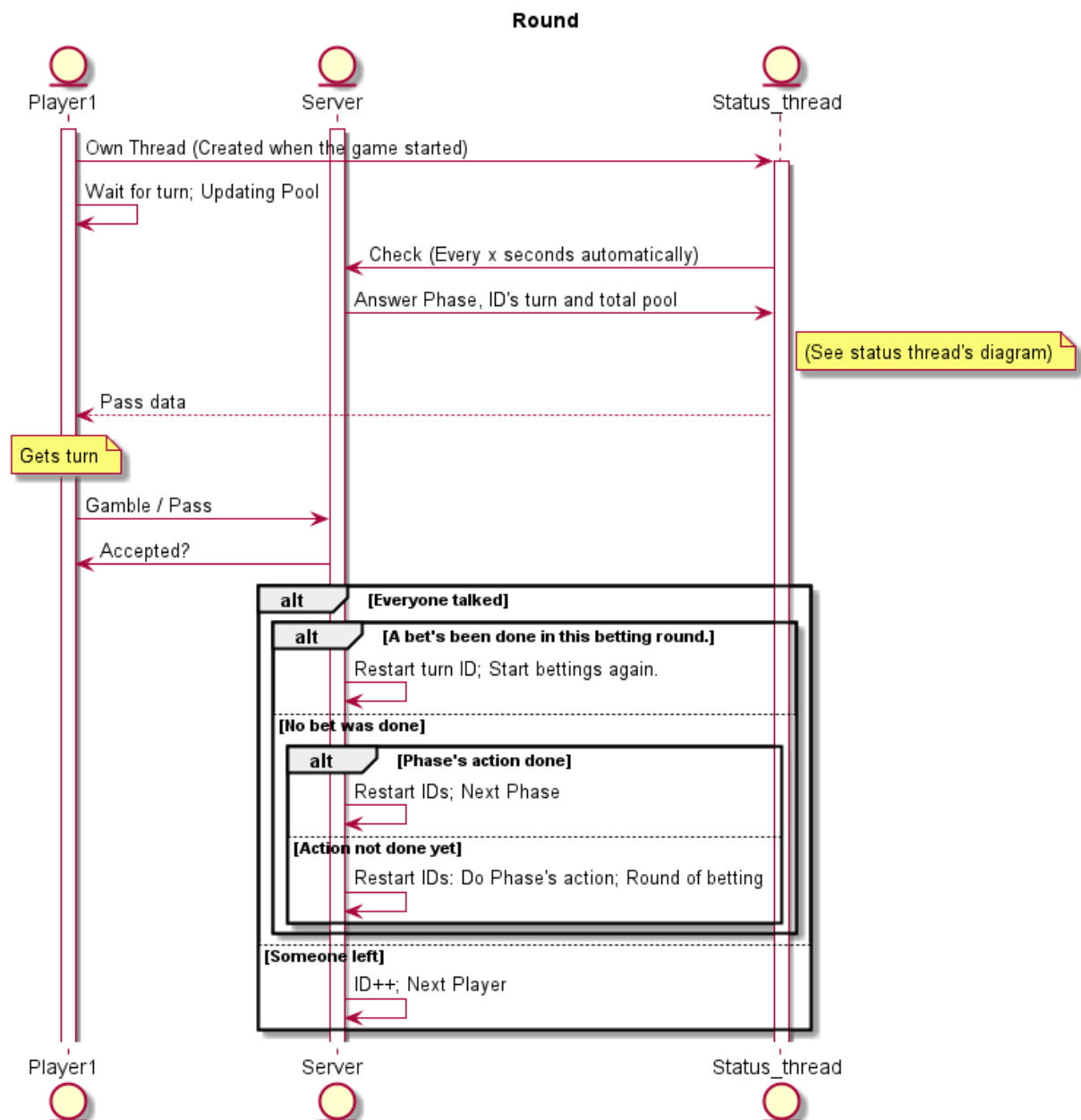
Thread independiente de estatus



Thread independiente que es lanzado por cada cliente en el momento de comienzo de la partida. Se encarga de realizar un ping cada 3 segundos al servidor para obtener la fase del juego, el ID de turno del jugador al cual le toca hablar y la cantidad de fichas actual en el pool común.

He decidido hacerlo de esta manera, ya que si sólo hubiera una única partida en juego a la vez no sería necesario el uso de *multithreading* en el programa. Pero como se soporta el uso de varias mesas de forma simultánea para no verme limitado por esta restricción, necesito algún sistema de control de estado de la partida.

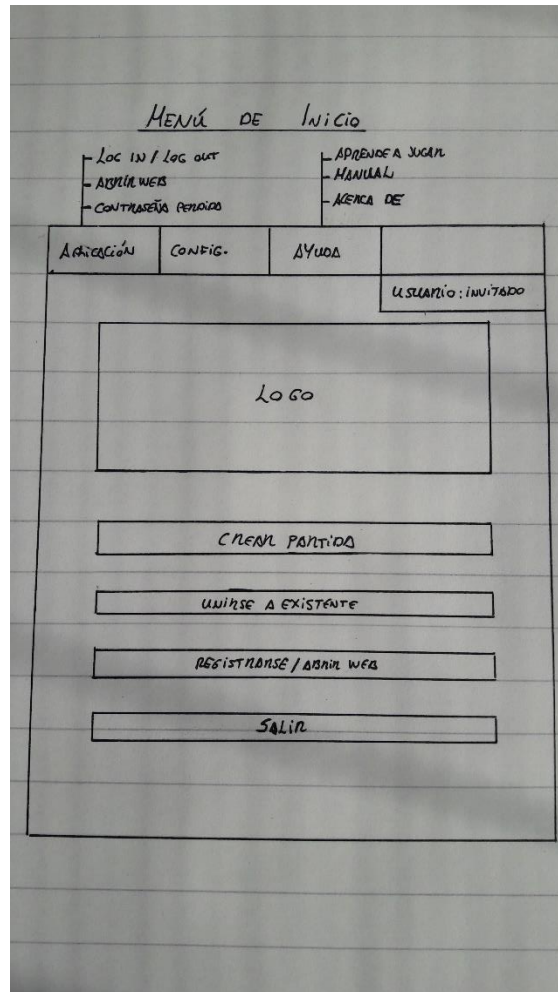
Funcionamiento de ronda



Comprobaciones a realizar tras cada acción realizada por cada jugador (de 1 a n). Se comprueba primero si todos han hablado, si queda alguno simplemente se pasa el turno al siguiente. Si han hablado todos se comprueba si alguno de ellos ha realizado una apuesta; Se seguirá repitiendo la fase actual de apuestas hasta que todos pasen o no suban la apuesta igual. Una vez hecho esto lo siguiente es comprobar si se ha realizado la acción propia de la fase, si no, se realiza y se vuelve a repetir otra fase de apuestas. Cuando ya se ha realizado y se cumplen todos los puntos anteriores, se pasa de fase y comienza la primera ronda de apuestas.

Análisis y diseño de la interfaz de usuario

Menú Principal



Es la ventana que se abrirá cuando el usuario inicie la aplicación. El logo es el propio de la aplicación y de fondo se mostrará uno relacionado con la temática del Poker. En la esquina superior izquierda muestra si el usuario se encuentra con sesión iniciada o está como invitado.

Botones (De izquierda a derecha y de arriba abajo)

- Barra de Herramientas:
 - Aplicación: Botones para iniciar sesión o cerrarla, así como para abrir el index de la página web o la relacionada para recuperar la contraseña en caso de pérdida.
 - Configuración: Distintas opciones de configuración propias de la aplicación.
 - Ayuda: Contiene dos secciones, una para mostrar las reglas y el funcionamiento de la progresión de fases en el Texas Hold'em y otra con una escala graduada de las diferentes manos posibles y su orden.
- Crear Partida: Crea una partida totalmente nueva.
- Unirse a partida: Se une a una partida previamente creada por otra persona.
- Registrarse / Abrir Web: En función de si el usuario se encuentra logeado, le abrirá la web en la página propia para registrarse, o le abre la web en el index.
- Salir: Cierra la aplicación.

Ventana de Juego

Ventana en la que se desarrolla la acción del juego.

Funcionamiento de botones:

- Barra de Herramientas:
 - Propiedades: Cierra la app. de forma adecuada para que el resto no lo tengan en cuenta.
 - Leyenda: Muestra las mismas pantallas de ayuda que se han explicado en el punto anterior.
- Destapar: Muestra las cartas correspondientes según la fase de juego actual.
- Apostar: Apuesta la cantidad indicada en el box adyacente.
- Pasar: Pasa su turno de apuesta; Equivalente a apostar 0 fichas.
- Retirarse: El jugador se retira de la ronda actual, el resto siguen jugando hasta finalizarla y cuando ocurre, este se reincorpora.

Explicación de áreas:

- Superior Izquierda: Información propia del Jugador. Incluye:
 - ID personal del Jugador. 'Invitado' si no se ha registrado o iniciado sesión.
 - Fichas restantes. Fichas personales del jugador que le quedan para la partida en curso.
 - Fase de Juego. Fase en la que se encuentra el juego, según ella se van realizando las distintas acciones propias intrínsecas a cada una.
- Superior Derecha: Información de conexión, muestra si hay problemas en la conexión del cliente al servidor.
- Superior Central: Información general.
 - Mi turno. Icono cambiante para indicar si es el turno de hablar del jugador.
 - Pool común. Cantidad conjunta de fichas apostadas que se lleva el ganador de la ronda.