Portal Games

Daniela Arenas Monroy

Juan José Cuervo Higuera

Juan Sebastián Jaramillo Silva

Mazo 2020.

Universidad de San Buenaventura.

Ingeniería de Sistemas.

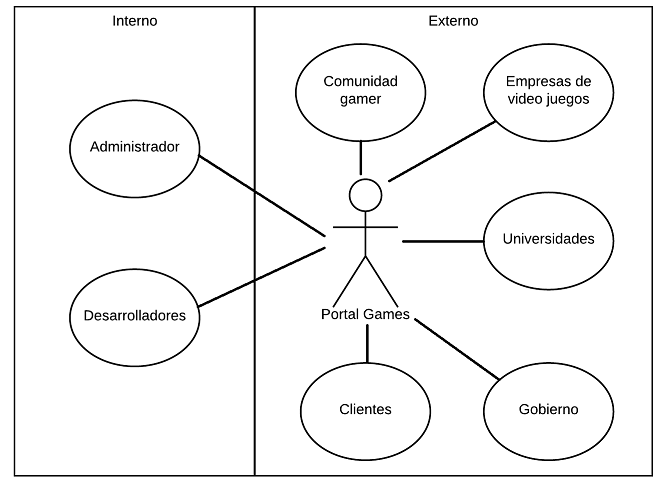
Arquitectura y patrones de Software

# Descripción del problema

# En las plataformas de videojuegos no hay un sistema que permita el acceso a una información detallada de las estadísticas del jugador respecto a varios juegos probados, de igual forma los usuarios no pueden compartir detalles de su progreso.

# Así mismo, no hay variedad de contenido en diferentes temáticas de tal forma que la experiencia del usuario no cumple las expectativas para jugar variedad de títulos.

**Stakeholders**



¿Cómo un programa puede mejorar la gestión de la información en plataformas de videojuegos mejorando la experiencia del usuario y permitiendo jugar diferentes temáticas?

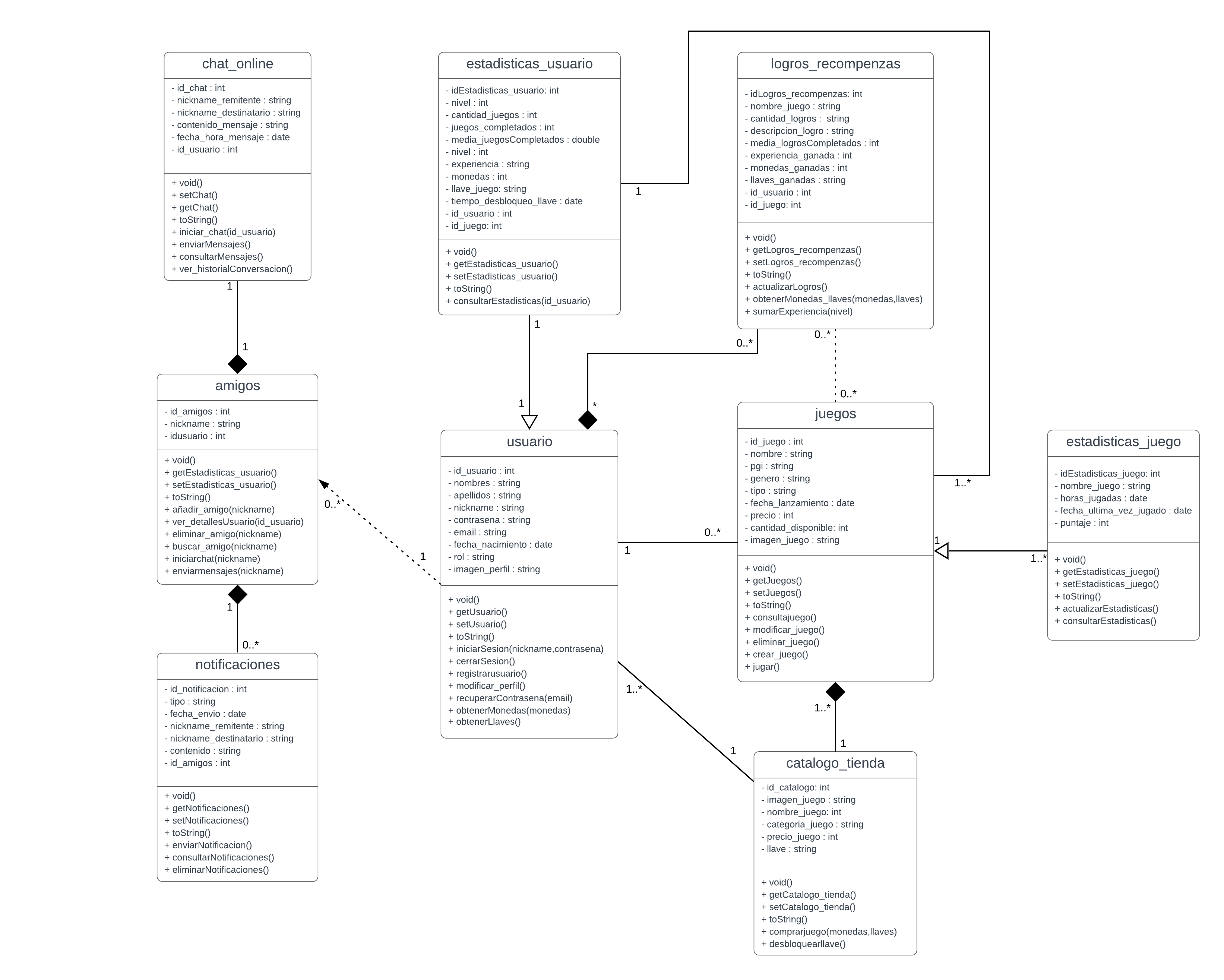
**Requerimientos Funcionales:**

|  |  |
| --- | --- |
| Requerimiento | Descripción |
| 1. Iniciar Sesión | Iniciar sesión mediante un usuario/correo y contraseña |
| 1. Registrar usuario | Llenar formulario de registro para crear usuario en la plataforma |
| 1. Recuperar contraseña | Mediante un código enviado por correo electrónico, se ingresa el código en la aplicación para modificar la contraseña |
| 1. Modificar perfil | Cada usuario puede modificar ciertos datos y opciones según convenga |
| 1. Consultar juegos en la tienda | Se cargará un listado filtrado por categoría de juego para comprar |
| 1. Comprar juegos | Dependiendo de las monedas que tenga el usuario, podrá comprar los juegos que desee |
| 1. Recargar monedas | Mediante un código ingresado por el usuario en la aplicación, o por parte del administrador, se le otorgan monedas al usuario |
| 1. Cargar juegos en biblioteca | Cuando el usuario compre un juego se añadirá a la biblioteca disponible para jugar |
| 1. Jugar juego en biblioteca | Mediante la selección del juego se añade |
| 1. Tablero de juego | Una vez se haya marcado un hito en un juego, se registrarán las estadísticas tales como: tiempo jugado, puntaje, nivel, experiencia. |
| 1. Ver estadísticas de usuario | Se registrará un listado de estadísticas tales como: juego, tiempo jugado, puntaje, nivel, experiencia de todos los juegos visitados por el usuario |
| 1. Recompensas | Por cada hito importante alcanzado en un juego, se recargarán monedas a la cuenta del usuario o el administrador publicará llaves únicas periódicamente el cual valen para desbloquear un juego gratis. |
| 1. Añadir amigos | Se consultará un listado de usuarios el cual se podrán añadir siempre y cuando el usuario añadido acepte la solicitud |
| 1. Ver estadísticas de amigos | Una vez 2 usuarios sean amigos se podrá ver las estadísticas individuales y se hará una comparativa de las estadísticas propias |
| 1. Chat local | Una vez 2 usuarios sean amigos se podrá activar una ventana de chat, el cual podrán conversar desde 2 computadores simultáneamente |
| 1. Enviar correo | Por cada “transacción” se enviará un correo electrónico a la cuenta registrada en la plataforma |

**Requerimientos no funcionales:**

|  |  |
| --- | --- |
| Requerimiento | Descripción |
| 1. Interfaz gráfica | El programa debe contar con interfaz gráfica para su interacción con el usuario. |
| 1. Manuales | El programa debe contar con manuales de usuario para aprender el uso de la plataforma |
| 1. Patrones | Se debe desarrollar el programa mediante patrones (Pizarrón y capas o cliente-servidor) |
| 1. Lenguaje de programación | Se debe desarrollar en el lenguaje de programación JAVA, siguiendo los estándares de patrones de arquitectura y estructuras de datos. |
| 1. Herramientas de desarrollo | Se debe usar el IDE Netbeans o Eclipse para el desarrollo de la aplicación |
| 1. Persistencia de datos | El programa debe contar con persistencia de datos en un motor de base de datos tales como MySQL, Oracle o MariaDB |
| 1. Usabilidad | Todas las interfaces deben ser de fácil manejo e intuitivas para el usuario final |
| 1. Contenido multimedia | El programa debe tener contenido multimedia para referenciar los juegos a los que se hacen alusión |

**Diagrama de clases**

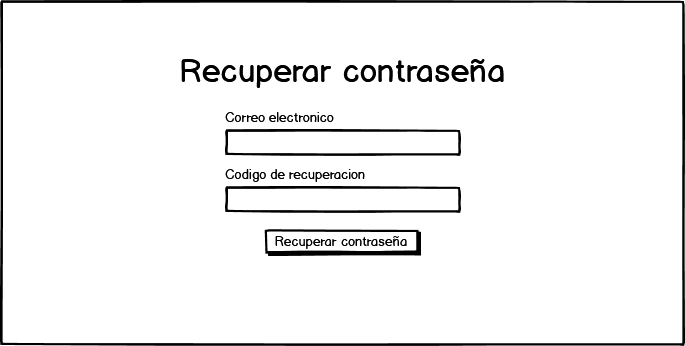


**Mockups:**

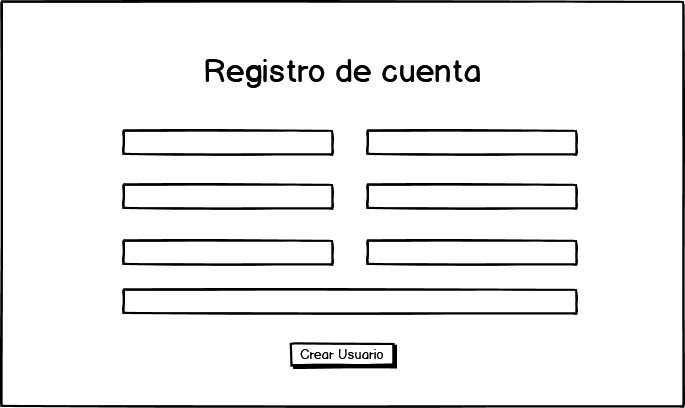
1. Login

****

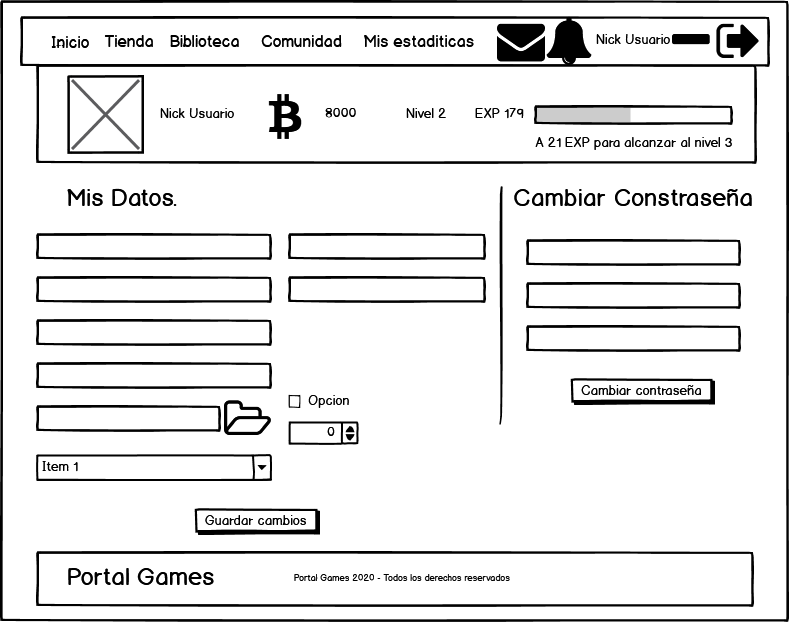
1. Recuperar contraseña

****

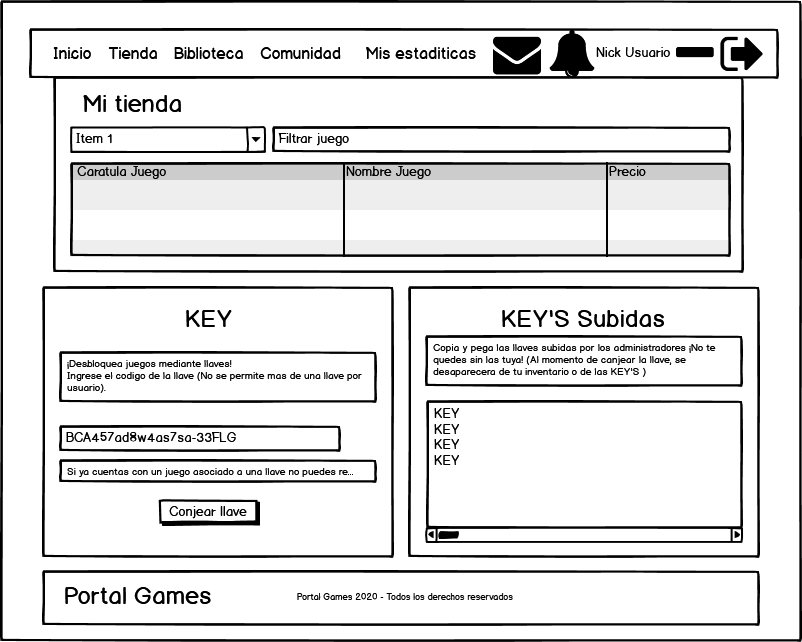
1. Registro cuenta

****

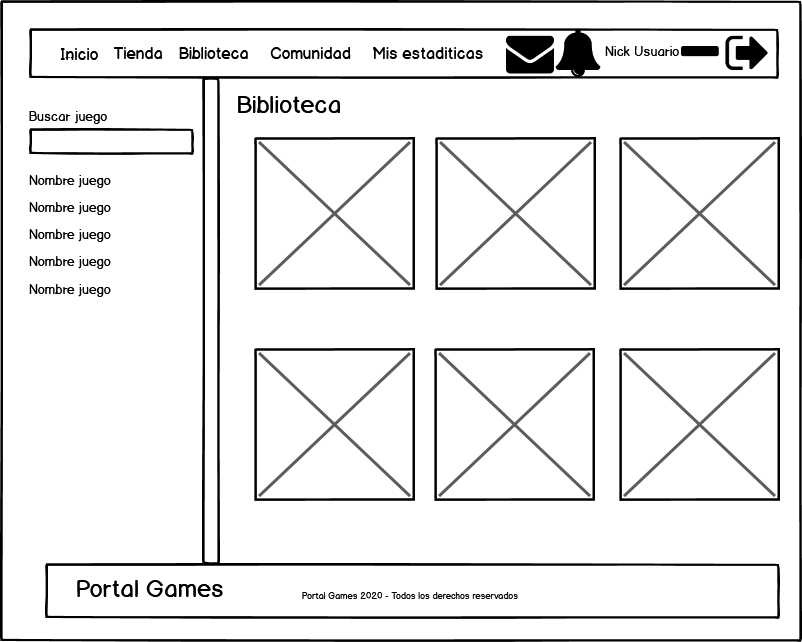
1. Inicio

****

1. Tienda

****

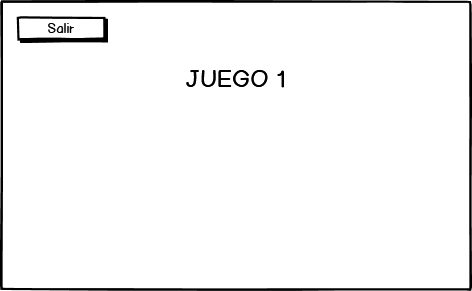
1. Biblioteca de juegos

****

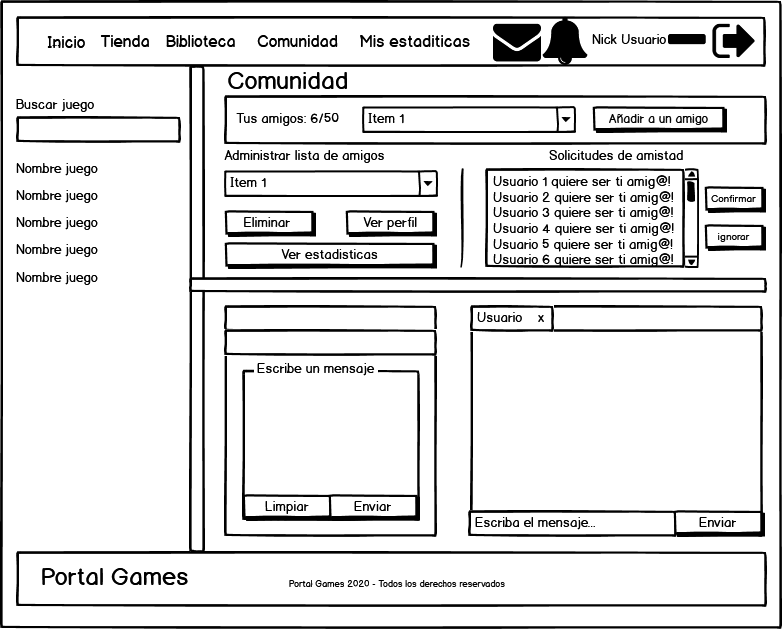
1. Detalles de juego

****

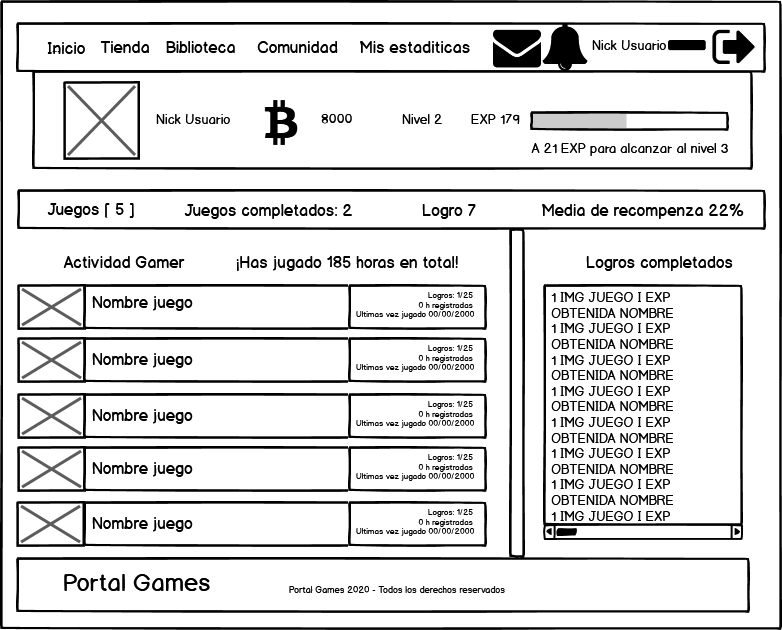
1. Juego

****

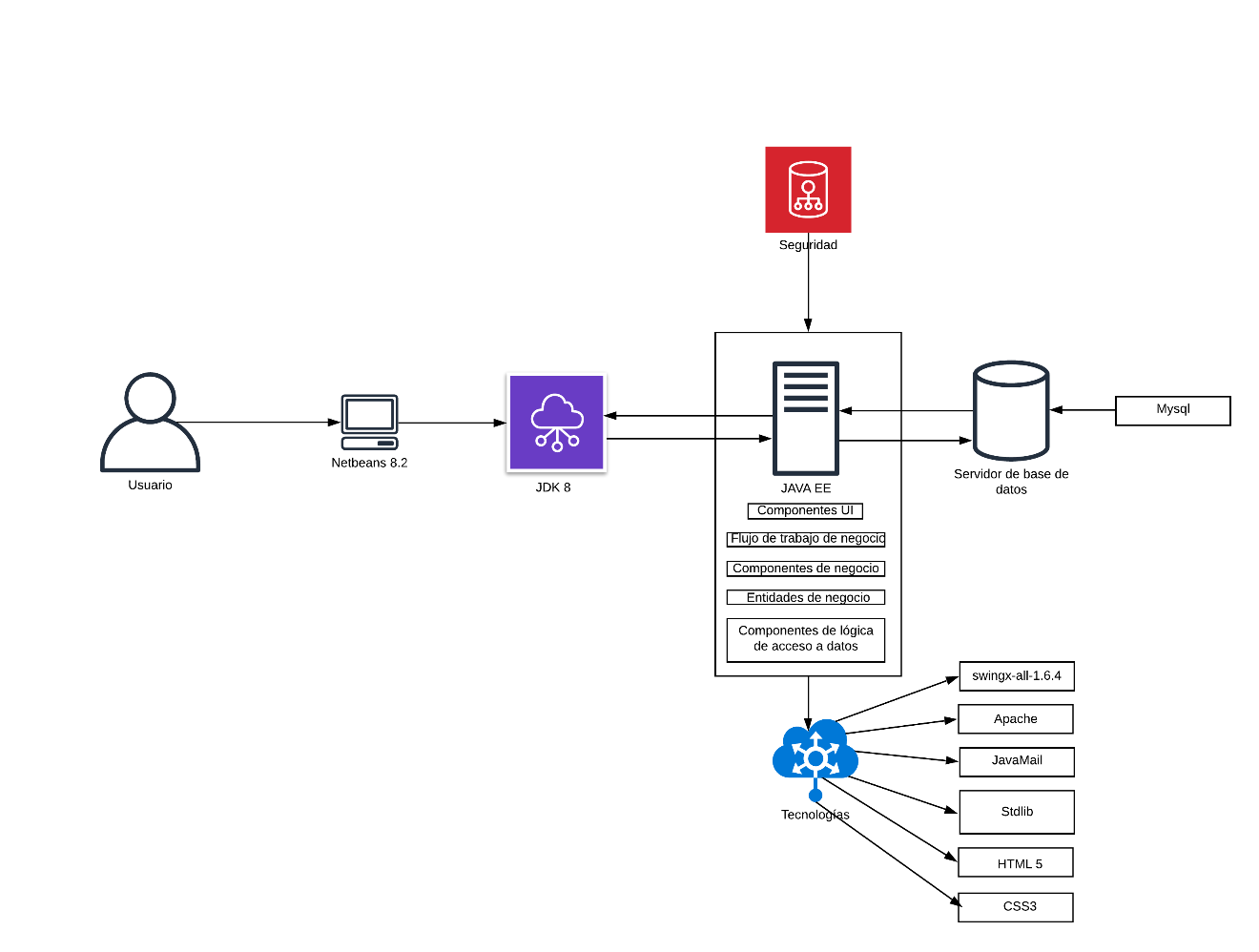
1. Comunidad

****

1. Estadísticas usuario

****

**Arquitectura de software**



**Imagen que contiene texto, mapa

Descripción generada automáticamente**

La arquitectura definida cliente-servidor comienza por un usuario, el cual accede a un cliente Windows y mediante el IDE Netbeans 8.2 obtiene los datos del servidor para su misma interacción. El servidor tiene componentes UI (gráficos), flujo de trabajo de datos, componentes de negocio, entidades de negocio y componentes de lógica de acceso a datos.

Los componentes de lógica de acceso a datos maneja tecnologías y librerías sirviendo como herramientas para el desarrollo de la aplicación, dichas herramientas son: swingx-all-1.6.4 (el cual permite el autocompletado de datos mediante un JCombobox), servidor local Apache, librería Java Mail (la cual permite la conexión de la aplicación por medio de un protocolo “smpt” a Gmail o hotmail), adicionalmente cuenta con la librería stdlib (la cual permite el desarrollo de juegos), finalmente cuenta con el lenguaje de etiquetas HTML y CSS3 para establecer una plantilla de diseño cuando llegan correos electrónicos generados desde la aplicación.

El servidor cuenta con encriptación de contraseñas, parametrización de funcionalidades asegurando principios de seguridad tales como la confiabilidad, confidencialidad e integridad.  
El servidor se conecta a un servidor de base de datos (Mysql) el cual es administrado por un gestor de base de datos MySQL Workbench 8.0 CE.

La arquitectura definida Pizarra representa las funcionalidades de la aplicación las cuales el usuario va a interactuar con cada una de ellas.

Evaluación:

|  |  |
| --- | --- |
| Criterio | Puntaje |
| Pertinencia / relevancia | 8 |
| Eficacia | 9 |
| Eficiencia | 8 |
| Impacto | 7 |
| Sostenibilidad | 9 |
| Comunicación | 10 |
| Creatividad | 8 |
| Diseño | 9 |
| Estilo arquitectónico | 10 |
| Funcionalidad | 8 |
| TOTAL | 86 |