



# Taller de sistemas de Información JAVA

Laboratorio Curso 2021 - Grupo 2

# Introducción y problema propuesto

- Sitio donde se publican videojuegos para la venta.
- Referencias: Steam, EpicGames.
- Usuarios clientes y desarrolladores conviven en la plataforma interactuando entre sí.
- Desarrolladores publican sus videojuegos.
- Jugadores compran, comentan, chatean, califican y descargan sus videojuegos.



# Requisitos Funcionales

- Los usuarios no registrados en el sitio (desde ahora llamados visitantes) podrán ver los juegos disponibles, también, acceder a ellos por medio de los menús de búsqueda (por categoría, nombre, últimas lanzamientos, tags y juegos en oferta).
- El visitante tendrá acceso a la lista de usuarios registrados, y podrá acceder a sus perfiles, así como también ver sus publicaciones en el muro.
- El visitante podrá registrarse en la plataforma, ya sea como Desarrollador o Jugador.
- Todos los usuarios registrados podrán loguearse en el sistema.
- El usuario registrado será identificado con un nickname y un email únicos en el sistema.



## Requisitos Funcionales

- Los jugadores podrán adquirir los juegos publicados en la plataforma.
- Los jugadores podrán abonar sus compras a través de PayPal.
- Los jugadores recibirán una factura con el detalle de la compra.
- Los jugadores podrán consultar sus adquisiciones en su biblioteca.
- Los jugadores podrán hacer publicaciones en su muro, acompañadas de imágenes (opcional).
- Los jugadores podrán calificar y comentar los juegos de sus propiedad.
- Los jugadores podrán reportar comentarios.
- Los jugadores podrán reportar videojuegos.
- Los jugadores podrán chatear con otros usuarios
- Los jugadores podrán gestionar su perfil.



## Requisitos Funcionales

- Los desarrolladores podrán publicar videojuegos en la plataforma
- Los desarrolladores podrán consultar sus ventas, cantidad de copias vendidas y monto recaudado.
- Los desarrolladores podrán agregar/quitar sus juegos de los eventos (ofertas) propuestos por la plataforma.
- Los desarrolladores podrán editar su perfil
- Los desarrolladores podrán consultar juegos bloqueados y solicitar desbloqueo.
- Los desarrolladores podrán hacer publicaciones en su muro.
- Los desarrolladores podrán usar el chat de la plataforma



# Requisitos Funcionales

- El administrador podrá crear y eliminar categorías.
- El administrador podrá crear eventos de ofertas (programar inicio y fin y descuento aplicado)
- El administrador podrá iniciar y detener los eventos creados manualmente.
- El administrador podrá acceder a estadísticas de la plataforma (ventas totales, ingresos, juegos más vendidos)
- El administrador podrá acceder a estadísticas específicas por juego y por desarrollador.
- El administrador podrá consultar, bloquear y desbloquear juegos y comentarios.

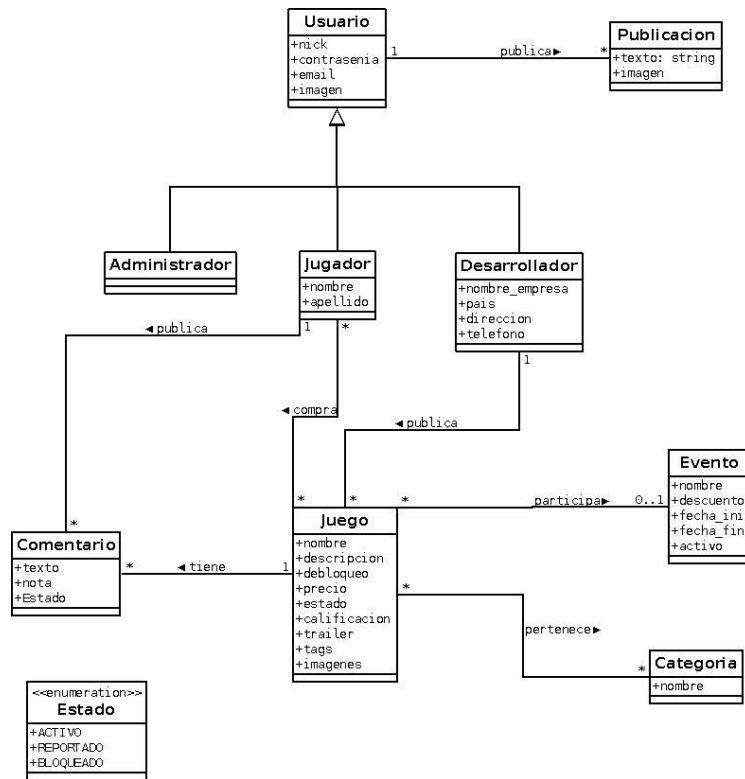


# Requisitos No Funcionales

- La aplicación será desarrollada utilizando el lenguaje Java y la plataforma de programación JavaEE.
- El front end de la aplicación está basado en estándares modernos
- Se utilizará un mecanismo de persistencia en base de datos MySql
- El frontend seguirá el patrón MVC.
- Se utilizará el framework Java Server Faces (JSF) para el desarrollo del frontend de la aplicación.
- El sistema debe proporcionar mensajes de error que sean informativos y orientados al usuario final.
- Las funcionalidades del front office serán accesibles desde cualquier dispositivo.

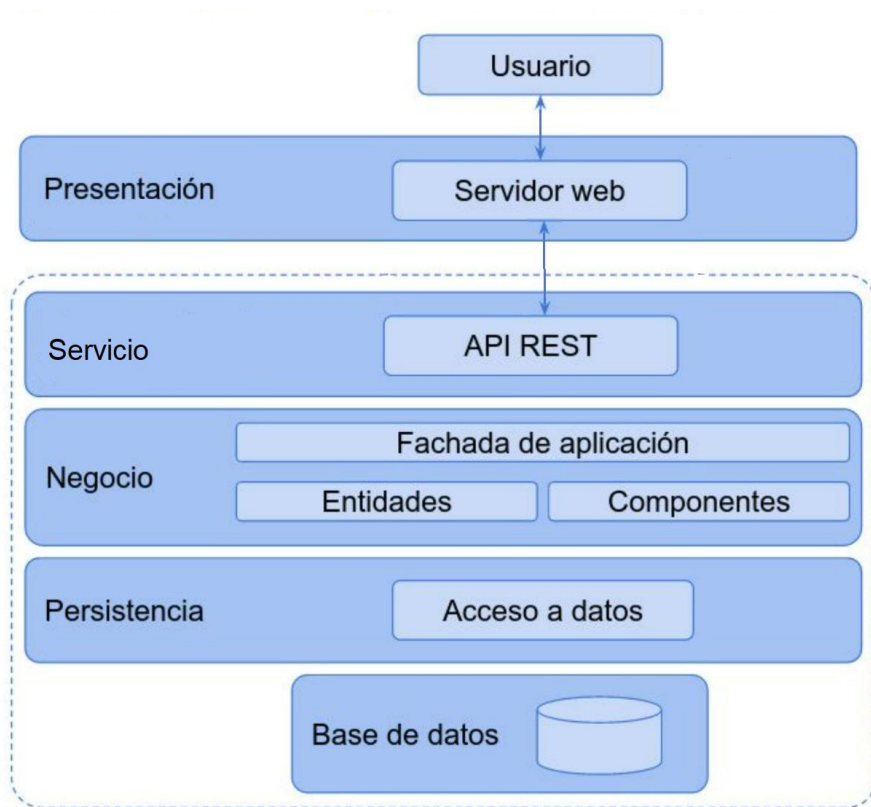
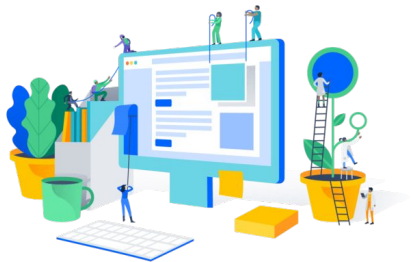


# Solución Planteada





# Arquitectura





## Tecnologías utilizadas

- El entorno de desarrollo utilizado fue Eclipse Enterprise Edition
- La implementación del proyecto se realizó con JavaSE 1.8.
- Para el manejo de datos y su mapeo a la base de datos se utilizó hibernate.
- Para la persistencia de datos utilizamos MySql.
- Se utilizó JSF en el frontend.
- El servidor utilizado para el despliegue de la aplicación es WildFly.
- Se utilizaron componentes de PrimeFaces y Bootstrap para los estilos.



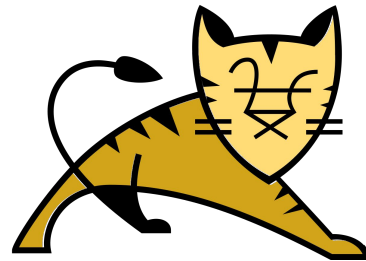
HIBERNATE





## Tecnologías descartadas

- Apache Tomcat, incompatibilidad con elementos de Java EE.
- Utilizamos RestEasy que funciona mejor con WildFly, por lo que descartamos Jersey, el cual nos venía dando algunos problemas.
- Optamos por MySQL sobre PostgreSQL para familiarizarnos con otro gestor de BBDD.



RESTful Web Services in Java.



PostgreSQL

# Aspectos a destacar

---

- Nuestro sistema tiene desarrollado la lógica de negocio como servicios.
- La interfaz de usuario se adapta a múltiples dispositivos. (Responsive)
- Es un sistema intuitivo y fácil de usar para el usuario.
- El panel del administrador cuenta con información estadística detallada y bien presentada



# Cosas a mejorar

---

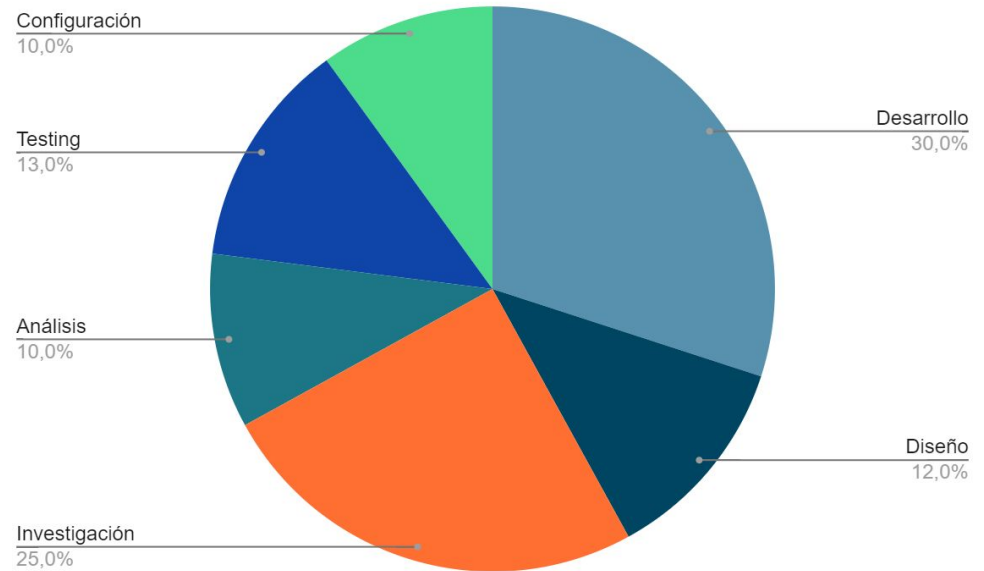
- Aspecto gráfico.
- Tiempos de cargas con margen de optimización.
- Mejorar la seguridad general.
- Estadísticas más completas para los desarrolladores.



# Distribución del trabajo

Total horas proyecto: 450

Distribuidas en 12 semanas.



# Problemas durante el desarrollo

- Pronunciada curva de aprendizaje para JSF y JavaEE.
- Entender la comunicación entre las vistas y los beans.
- Aprender el manejo y funcionamiento de un framework (primera vez).
- Configuración del entorno de trabajo. (Instalacion, versiones, librerías, incompatibilidades, etc)
- Envío de archivos a Rest API.
- Integración con método de pago Paypal.
- Adaptación al nuevo IDE (Eclipse Enterprise).
- Uso de GitHub.



## Trabajos a futuro o ampliación del sistema

- Suscripción a la plataforma con cartera de juegos incluida, regalos mensuales, etc
- Otros métodos de pago
- Agregar la posibilidad de hacer y ver streaming de otros jugadores.
- Implementar una app para móviles
- Lista de deseos
- Envío de DM entre los usuarios
- Inicio de sesión utilizando redes sociales
- Notificaciones de eventos a los jugadores.

