

Funcionalidad Volátil en Juegos Web: Una Arquitectura de Soporte

Leandro Manuel Pérez

Facultad de Informática. U.N.L.P.

23 de Noviembre 2009

Objetivos y Metodología

Objetivos

- Tema de estudio
- Desarrollos
- Proceso
- Resultados

Metodología

- Recorrido cronológico sobre el desarrollo
- Conceptos estudiados
- Resultados obtenidos
- Pruebas en vivo

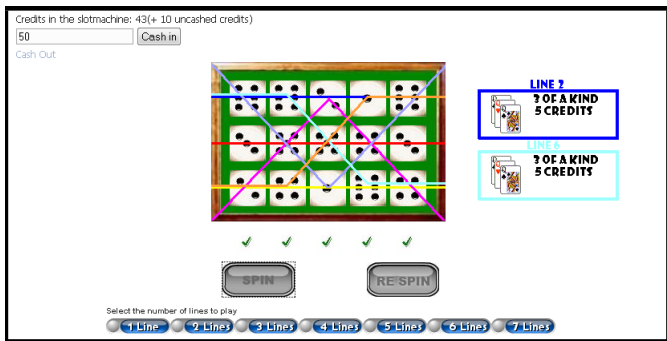
Comienzo

- Abril 2007.
- Lifa + ID Interactive.
- Innovación en Software de Entretenimiento.
- Alternativas web para juegos de casino.

Slotmachines



Prototipo desarrollado



Objetivos

- Juego Web.
- Evaluar nueva lógica de respin.
- Aspectos matemáticos.

Demostración del prototipo

Metodología de trabajo

Refactoring

- Desarrollo incremental
- Refactoring constante.

Testing

- Funcionalidad: Líneas, Spin, Respin, Cash In, etc.
- Lógica compleja de apuestas.
- Persistencia de eventos.
- Recuperación frente a caídas.

Aplicación Casino

- Contenedor de juegos
- Manejo de usuarios y cuentas



Side Games

- Juego secundario, dependiente, desacoplable.
- Objetivo: Side games en juegos reales.



Doble o Nada

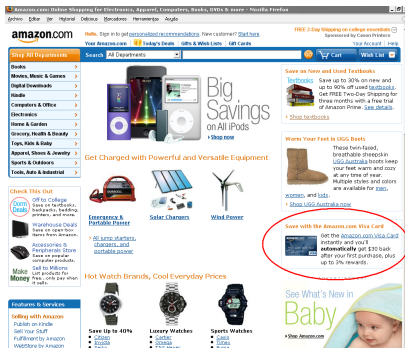
Características

- Altera la lógica del juego principal.
- Sienta las bases para la construcción de un pequeño FWK.
- Buen caso de funcionalidad volátil.

Funcionalidad volátil

- Caracter temporal.
- Requerimientos volátiles.

Ejemplo



Impacto

- Vista.
- Navegación.
- Modelo.

Problemas

- Impacto en Aplicación Host.
- Diseño.
- Código.
- Bugs.
- Mantenimiento.
- Costos.
- Deterioro de la calidad.

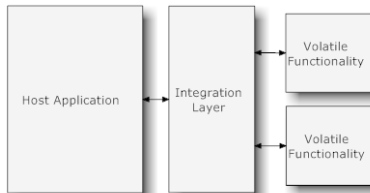
Características

- Volatility Pattern.
- Extent.
- Intent.

Aproximación

- Modelar funcionalidad volátil.
- Diseño e implementación separados.
- Integración.
- Obliviousness.
- Decoupling.

Arquitectura



Objetivo de la tesis

- Estudiar F.V. en Smalltalk y Seaside.
- Estudiar F.V. con problemas de comunicación complejos.
- Encontrar una solución a side games.

Ejes de la arquitectura

- Diseño e implementación por separado.
- Integración transparente.
- Comunicación bidireccional.
- Reutilización de la funcionalidad volátil.

Side Game 2: Card Wars

- Creación de componenetes genéricos, surge un pequeño FWK.
- Refactorización de DoN.
- Pensado para funcionalidad volátil.
- Surgimiento de una Arquitectura

Refactoring

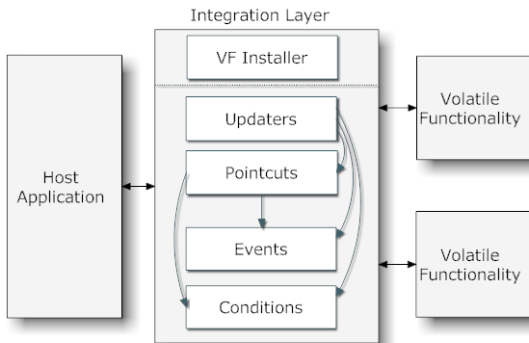
Primer Etapa

- Limpieza: Game, Side Games, Side Games FWK.
- Events
- Updaters
- Conditions

Segunda Etapa

- Limpieza: Instalación y elementos de capa intermedia.
- Pointcuts

Componentes



Updaters

- Notifier
- Target
- Events
- Condition
- Actions

Demo

Informe Final

- Documentación continua.
- Trabajo relacionado hacia el final de la implementación.

Arquitectura

- Características

Caso de Funcionalidad volátil en Smalltalk

- 1 Juego principal
- 2 Side Games como Funcionalidad volátil.
- Sistema integrador.
- Aplicación de configuración de Funcionalidad Volátil.

Trabajo relacionado

- Estudio extenso sobre trabajo relacionado
- 8 Fuentes

Trabajo pendiente

- Integración de F.V. en base de datos.
- Composición avanzada de GUIs.
- DSL para definición de Volatility y Connexion Pattern.
- Interacción entre F.Vs.

Conclusiones

- Implementación funcional.
- Arquitectura.
- Análisis del problema.
- Estudio de trabajo relacionado.
- Informe completo.

Agradecimientos

- Flia
- Romi
- Federico Balaguer
- Gustavo Rossi
- Max