

# Orientacion a Objetos 2

Dra Alejandra Garrido

Dr. Alejandro Fernandez

Dr. Gustavo Rossi



[garrido, gustavo, alejandro.fernandez]@lifia.info.unlp.edu.ar



Laboratorio de Investigación y Formación en Informática Avanzada  
Facultad de Informática, Universidad Nacional de La Plata  
Calle 50 esq. 115 - Primer piso (1900) La Plata, Argentina  
TE: (0221) 422 8252 <http://www-lifia.info.unlp.edu.ar>

- ✓ Si están inscriptos en la materia (vía Guaraní) ya tienen acceso al curso
  - ✓ (OO2 – 2016) **Orientación a Objetos II**
- ✓ Vamos a utilizar la plataforma para publicar
  - ✓ Todo el material utilizado en clase
  - ✓ Todas las prácticas y actividades
  - ✓ Consultas on-line (foros)
  - ✓ Comunicaciones de la cátedra
  - ✓ Horarios (o cambios en los mismos)

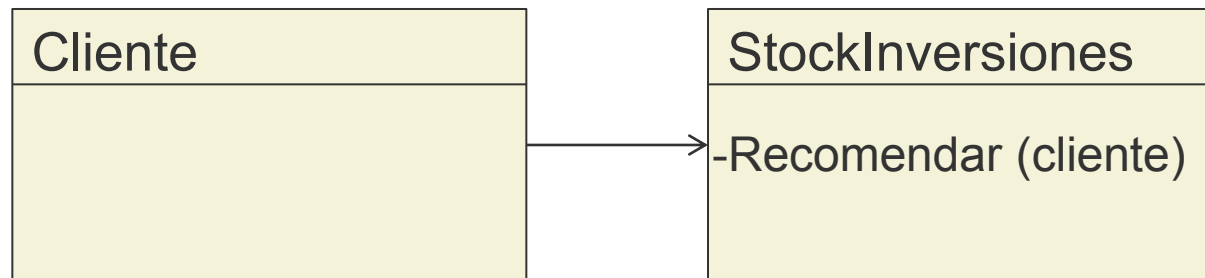
- ✓ Todos los horarios y aulas están publicados en la plataforma
- ✓ Inicia el martes 27/3, 8:00, en el horario de explicación de práctica

# Problemas en el diseño con objetos

- ✓ Encontrar las clases adecuadas
- ✓ Identificar correctamente las responsabilidades (metodos)
- ✓ Diseñar para el cambio (incluso para el no previsto)
- ✓ Buscar diseños modulares

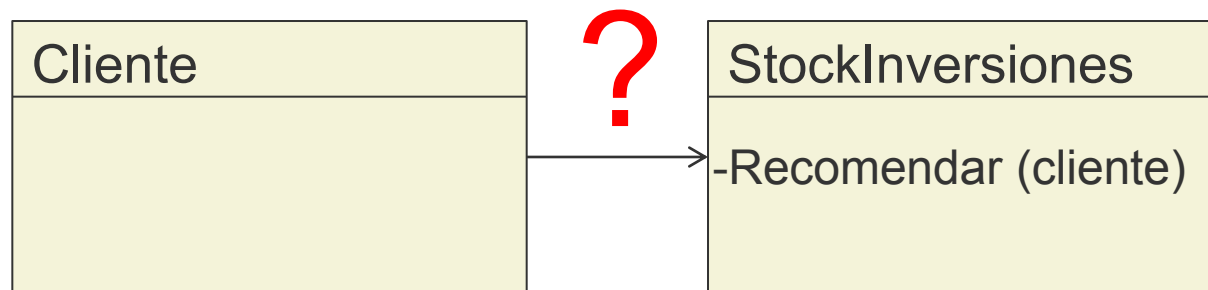
# Ejemplo

- ✓ Un sistema para asesorar clientes de un banco para inversiones
- ✓ Luego de un extenso analisis (y comprension del dominio) llegamos a un diseño...



# Evolución de la aplicación

- ✓ No hay una sola forma de recomendar (distintos tipos de analisis financiero)
- ✓ El algoritmo de recomendacion puede que dependa del “perfil” del cliente, o del momento del año, o de la situacion economica



# Solucion?

- ✓ Hacer diseños mas sensibles a la evolucion
- ✓ Como me doy cuenta que necesito en este caso particular?
- ✓ Con mas experiencia...mia, o de otros

**Patrones de Diseño**

# Como desarrollamos?

- ✓ Cuanto tiempo nos puede llevar diseñar un sistema ultra-flexible?
- ✓ Es aceptable para los clientes?
- ✓ Hay alternativas?

**Metodos Agiles**



# Somos Agiles...y entonces?

- ✓ Los metodos agiles suelen buscar resultados “rapidamente”, para chequear requerimientos con los clientes
- ✓ Pero los diseños entonces quedan “feos”

Refactoring



# Que pasa con la “correctitud” del sistema

- ✓ Somos agiles y ademas...
- ✓ Nuestros diseños son modulares (soportan el cambio)
- ✓ Pero como sabemos que el sistema funciona? Esperamos hasta...cuando?

**Test Driven Development (TDD)**

# Interfaces, Persistencia...

- ✓ Queremos que nuestro sistema muestre informacion en distintas formas (barras, estadisticas, mapas....)
- ✓ Y queremos guardar los datos en diferentes formatos....
- ✓ Pero todo esto no tiene que ver con nuestro problema....
- ✓ Como nos “conectamos” con otras clases

## Frameworks