

Grupo 4:

- Llauro, Manuel
- Inoriza, Pablo
- Arturi, Augusto
- Blanco Sebastian Ezequiel

Instrucciones para correr el programa:

Montar el servidor

```
$ cd offers_validator  
$ lein deps  
$ lein ring server-headless
```

Luego para ejecutar la interfaz ~/gui/**StartGui.java**

Patrones de diseño utilizados

MVC En la vista: supervisor y cajero.

Facade para hablar entre vista y el controller (interface segregation), nos fue muy util para abstraer al front-end del modelo, sabiendo que ésta únicamente tenía que interactuar con la interfaz *CashRegisterInterface* permitiéndonos distribuir el trabajo entre varios integrantes.

State para los estados de la *CashRegisterState* (clase abstracta) la cual se implementa en las clases *Open* y *Close*

Singleton para loguear la información del ticket de control, con la finalidad de instanciar solo una vez al objeto.

Adapter este patrón nos fue muy útil para adaptar la implementación, al comenzar a implementar se pactó que el los parámetros fuesen de determinada forma entre 2 de los integrantes para distribuir el trabajo, al finalizar hubo cambios, con el siguiente patrón se logró adaptar el código de manera tal de recibir lo esperado.

Modelo del dominio:

