

Nom:

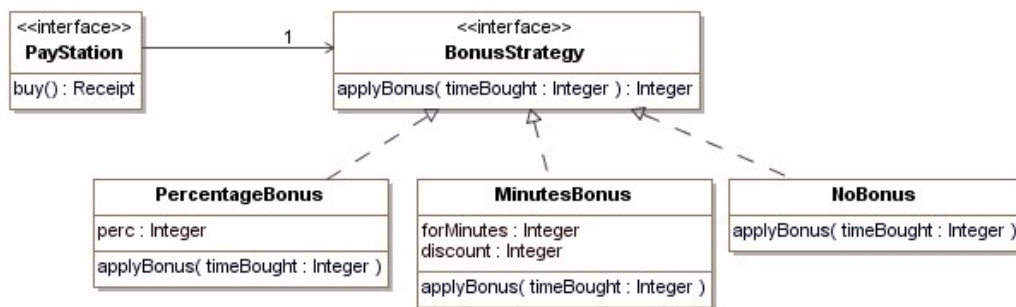
Ens han demanat afegir un nou requisit funcional al software PayStation (que hem estat desenvolupat a la classe de laboratori) per tal de que els parquímetres puguin aplicar bonificacions en el temps que els seus clients compren. Aquesta bonificació es calcularà després de que el client hagi introduït totes les monedes i el ticket contindrà el temps comprat incloent la bonificació.

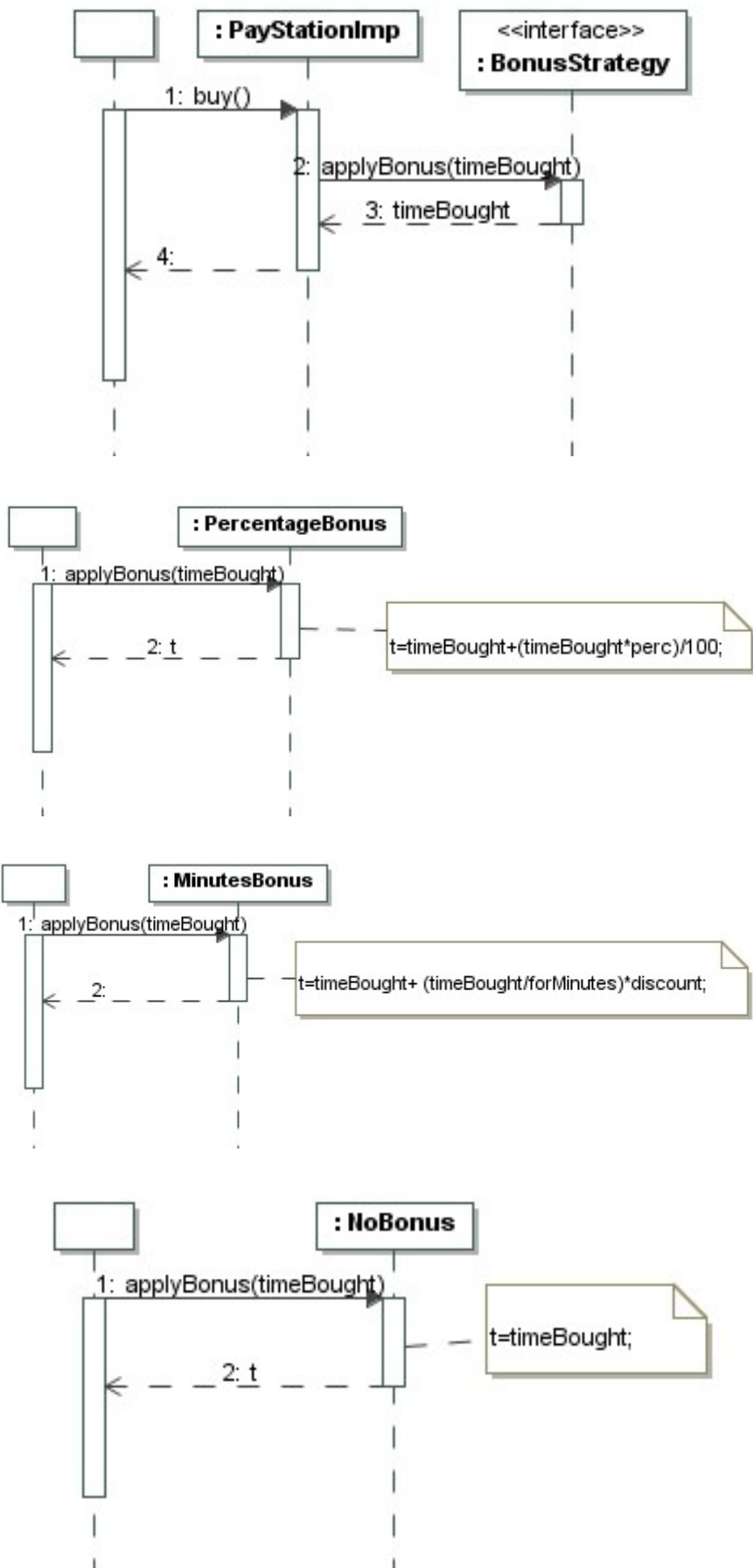
Inicialment, el software oferirà tres maneres diferents de bonificar aquest temps (inclosa la no bonificació). Cada parquímetre podrà seleccionar la manera de bonificar als seus clients:

- 1) *Bonificació en percentatge minuts*. S'incrementaren els minuts comprats en el percentatge de la bonificació. Per exemple, si un client ha comprat 10 minuts i el percentatge de bonificació és del 10% llavors els minuts totals comprats, incloent la bonificació, seran 11 minuts. Si el resultat d'aplicar la bonificació no dóna un nombre de minuts enter, s'aplicarà l'arrodoniment per baix.
- 2) *Bonificació en minuts*. Per una quantitat determinada de minuts comprats s'aplicarà una bonificació en un nombre de minuts determinat. Per exemple, si per cada 5 minuts comprats es bonifica amb 1 minut addicional, si comprem 10 minuts, els minuts finals obtinguts, incloent la bonificació, seran 12 minuts. Si el nombre minuts sobre els que s'ha d'aplicar una bonificació és més petit que els minuts bonificats es retornarà l'excepció *BonificacióIncorrecta*. Per exemple, si el parquímetre bonifica en 10 minut per cada 5 minuts s'activarà l'excepció.
- 3) *Sense bonificació*. No s'aplicarà cap bonificació als minuts comprats.

1. [3 punts] (Competència transversal) Expliqueu de forma **detaillada i raonada** quina solució proposaríeu per afegir el nou requisit. Descriviu la vostra solució tant des del punt de vista estàtic (diagrama de classes) com des del punt de vista del comportament (diagrama de seqüència). Volem una solució que: 1) permeti afegir noves maneres de bonificació (sense fer canvis, només extensions) i 2) permeti als parquímetres canviar la manera de calcular les bonificacions en temps d'execució. **Contesteu aquesta pregunta a l'espai que teniu disponible.**

Utilitzaríem una solució on aplicariem el patró estratègia. El comportament variable de l'aplicació de la bonificació quedaria encapsulada en una interface i les classes que implementarien aquesta interface implementarien el càlcul concret que han de tenir els parquímetres. A continuació es mostra el diagrama de classes per aquesta solució:





2. [3 punts] Expliqueu clarament les iteracions que seguireu per implementar aquest nou requisit al software PlayStation utilitzant TDD. Cal que quedi clar el que fareu a cada iteració. **Contesteu aquesta pregunta a l'espai que teniu disponible.**

Iteració 1: (Refactoring) – L'operació buy ja existia i estava treballant amb una estratègia NoBonus.

- Afegir a l'operació buy de PayStationImpl la referència a la interface BonusStrategy i usar la referència per calcular la bonificació.
 - Afegir la classe (NoBonus) que implementi la interface BonusStrategy.
 - Introduir la interface BonusStrategy
 - Refactoritzar l'operació de setUp per tal de que en crear l'objecte PayStationImpl usi l'estratègia no bonus.
- Iteració 2 (Percentage Bonus): Definiu un nou test case, TestPercentageBonus per definir els tests relacionats amb el mètode de bonificació per percentatge.
 - Definiu el test per provar aquesta bonificació, creant una instància de l'estratègia amb un percentatge de 10. Apliquem la bonificació a 6 minuts (cas arrodoniment per baix que ens ha de donar 6 minuts amb bonificació inclosa).
 - Definiu el codi per tal de que el test anterior passi.
- Iteració 3 (Percentage Bonus):
 - Definiu el test per provar aquesta bonificació, creant una instància de l'estratègia amb un percentatge de 10. Apliquem la bonificació a 10 minuts (cas no arrodoniment que ens ha de donar 11 minuts amb bonificació inclosa).
 - Definiu el codi per tal de que el test anterior passi.
- Iteració 4 (Minuts Bonus): Definiu un nou test case, TestMinutsBonus per definir els tests relacionats amb el mètode de bonificació per minuts.
 - Definiu el test per provar aquesta bonificació, creant una instància de l'estratègia amb un bonificació per minuts 5,1 (escenari de l'enunciat). Apliquem la bonificació a 4 minuts (cas excepció, minuts a bonificar < bonificació en minuts).
 - Definiu el codi per tal de que el test anterior passi.
- Iteració 5 (Minuts Bonus):
 - Definiu el test per provar aquesta bonificació, creant una instància de l'estratègia amb un bonificació per minuts 5,1 (escenari de l'enunciat). Apliquem la bonificació a 10 minuts (cas bonificació on els minuts totals a bonificar és múltiple dels minuts del client, ens ha de donar 12).
 - Definiu el codi per tal de que el test anterior passi.
- Iteració 6 (Minuts Bonus):
 - Definiu el test per provar aquesta bonificació, creant una instància de l'estratègia amb un bonificació per minuts 5,1 (escenari de l'enunciat). Apliquem la bonificació a 6 minuts (cas bonificació on els minuts totals a bonificar no és múltiple dels minuts del client, ens ha de donar 7).
 - Definiu el codi per tal de que el test anterior passi.
- Iteració 7:
 - Definir un test case TestIntegration amb 2 tests per provar que els pay station funciona amb les dues estratègies de bonificació tal i com s'espera.