

Pràctica 3: Proves Unitàries

Enginyeria del Programari

Alumne: Roger Arqués Vall

Professor: Juan Manuel Gimeno Illa

Data d'entrega: 23/01/2017

Curs 2016/2017

Índex

1. Introducció	2
2. Tests del paquet “data”	3
3. Tests de “ActivationCard”	4
4. Tests de “VotingMachine”	5
4.1. Tests de <i>activateEmission()</i>	5
4.2. Tests de <i>canVote()</i>	5
4.3. Tests de <i>vote()</i>	6
4.4. Tests de <i>sendReceipt()</i>	6
5. Dobles dels serveis col·laboradors	7

1. Introducció

L'objectiu d'aquesta pràctica era implementar una simplificació del cas d'ús "*Emit Vote*", treballat durant tota l'assignatura, i crear un conjunt de tests per assegurar, en la mesura del possible el seu correcte funcionament.

Aquest document s'explicarà en llenguatge natural la situació que ha provar cada un dels diferents tests així com els dobles utilitzats per als serveis col·laboradors.

2. Tests del paquet “data”

En el paquet “data” hi ha situades les classes valor *MailAddress*, *Party*, *Signature*, *Vote* i *IrisScan* corresponent a la part opcional de la pràctica.

Els tests sobre aquestes classes estan situats al directori *src/test/java/data* i són molt senzills. Cada classe de test **Test.java* té 2 tests, un per provar el mètode *equals()* i un altre per provar *hashCode()*. Bàsicament els tests comproven que dos objectes iguals tinguin el mateix *hashCode* i el seu mètode *equals()* retorni *True*.

3. Tests de “ActivationCard”

Els tests sobre la classe *ActivationCard* estan situats al directori *src/test/java/kiosk*. La *ActivationCard* es construeix al mètode *@Before*.

- Test 1 - *newActivationCardIsActive()*: El test comprova que una nova *ActivationCard* està activa just després de crear-se.
- Test 2 - *activationCardCreatedWithCorrectCode()*: Comprova que la targeta s'hagi creat amb el codi proporcionat
- Test 3 - *erasedActivationCardIsNotActive()*: Comprova que una tarjeta amb el codi esborrat està desactivada.
- Test 4 - *erasedCodeIsNull()*: Comprova que, un cop esborrat el codi de la tarjeta, aquest val *null*.
- Tests 5 i 6 - *testEqualsDifferentCodes()*, *testEqualsSameCodes()*: Comproven el funcionament del mètode *equals* per tarjetes amb el mateix codi i amb diferent codi respectivament.

4. Tests de “VotingMachine”

Hi ha 4 classes de test dedicades a la classe *VotingMachine*, una per cada mètode a provar. En cada classe de test, en el mètode *@Before*, s’inicialitza una nova *VotingMachine* i se li injecten tots els dobles necessaris dels sistemes col·laboradors.

4.1. Tests de *activateEmission()*

- Test 1 – *correctlyActivatedEmission()*: Es comprova la correcta activació de la *VotingMachine* amb un tarjeta i un codi d’activació vàlids.
- Test 2 – *errorActivatingEmissionInvalidCode()*: Si s’intenta activar la *VotingMachine* amb un codi invàlid, s’ha de llençar un error.
- Test 3 – *cannotAcivateTwice()*: Si s’intenta activar la *VotingMachine* dues vegades seguides sense vota s’ha de llençar un error.
- Test 4 - *activateEmissionWithIrisScan ()*: Si la *VotingMachine* s’activa amb un escaneig d’iris, el codi de l’iris i el de la tarjeta han de ser iguals per activar-se.
- Test 5 – *activateEmissionWithIrisScanCodesNotMatching()*: Es comprova que en un intent d’activar la *VotingMachine*, si el codi de l’iris i el de la tarjeta no són iguals s’ha de llençar un error.

4.2. Tests de *canVote()*

- Test 1 – *voterCanVote ()*: Es comprova que si la *VotingMachine* s’activa correctament el mètode *canVote()* ha de retornar cert.
- Test 2 – *errorActivatingEmissionInvalidCode()*: Si la *VotingMachine* no s’ha activat prèviament, *canVote()* ha de retornar fals.

4.3. Tests de *vote()*

- Test 1- *voteRegisteredCorrectly()*: Comprova que es pugui votar correctament si la *VotingMachine* s'activa amb una *ActivationCard* vàlida prèviament.
- Test 2 - *cannotVoteIfMachineNotActivated()*: Si s'intenta votar sense haver realitzat l'activació prèviament s'ha de llençar un error.
- Test 3 - *cannotVote2Times()*: Si s'intenta votar dos cops sense reactivar la *VotingMachine* s'ha de llençar un error.
- Test 4 - *codeFromActivationCardErased()*: Es comprova el correcte esborrat del codi de la *ActivationCard* un cop s'ha votat.
- Test 5 - *voteIsPrintedOneTime()*: Es comprova que per cada vot s'imprimeixi exactament un comprovant físic.
- Test 6 - *consecutiveVaildVotes()*: Es comprova que dos votants diferents amb dues *ActivationCards* diferents però vàlides puguin votar consecutivament.

4.4. Tests de *sendReceipt()*

- Test 1 – *cannotSendIfMachineNotActivated ()*: Es comprova que no es pugui enviar el comprovant electrònic si no s'ha activat la *VotingMachine*.
- Test 2 – *cannotSendIfNotVoted ()*: Es comprova que no es pugui enviar el comprovant electrònic si no s'ha votat prèviament.
- Test 3 – *sentToCorrectAddress()*: Es comprova que el comprovant s'envia a l'adreça proporcionada.
- Test 4 – *sentCorrectSignature()*: Es comprova que la signatura del comprovant enviat sigui la que retorna el *SignatureService*.

5. Dobles dels serveis col·laboradors

S'han diversos dobles per a cada un dels serveis col·laboradors per tal de poder comprovar situacions en les quals els serveis poden retornar diferents respostes, com per exemple, que no s'ha pogut validar la *ActivationCard* o just el contrari, que s'ha validat correctament.

Per cada servei s'ha creat un doble *Forbidden* en el qual tots els mètodes llencen una excepció. Serveixen per prevenir la crida d'alguns mètodes en determinats tests.

La resta de dobles son els dobles més senzills possibles, només saben retornar una resposta per cada mètode.