s

BLACK BOXING

Classe id\_avviso:

CE1 : [0, MAXINT] --------------------------------------------------------------------------------------------------------VALIDA

CE2 : [MININT, -1] -----------------------------------------------------------------------------------------------------NON VALIDA

Classe idUtente:

CE1 : [0, MAXINT] --------------------------------------------------------------------------------------------------------VALIDA

CE2 : [MININT, -1] -----------------------------------------------------------------------------------------------------NON VALIDA

STRATEGIA SECT

NUMERO TEST CASE : 2 \* 2 \* 3 = 12

TC1 : (10, 10) CE1/CE1

TC2 : (0, 0) CE1/CE1 estremi

TC3 : (MAXINT, MAXINT) CE1/CE1 estremi

TC4 : (10, -10) CE1/CE2

TC5 : (0, MININT) CE1/CE2 estremi

TC6 : (MAXINT, -1) CE1/CE2 estremi

TC7 : (-10, 10) CE2/CE1

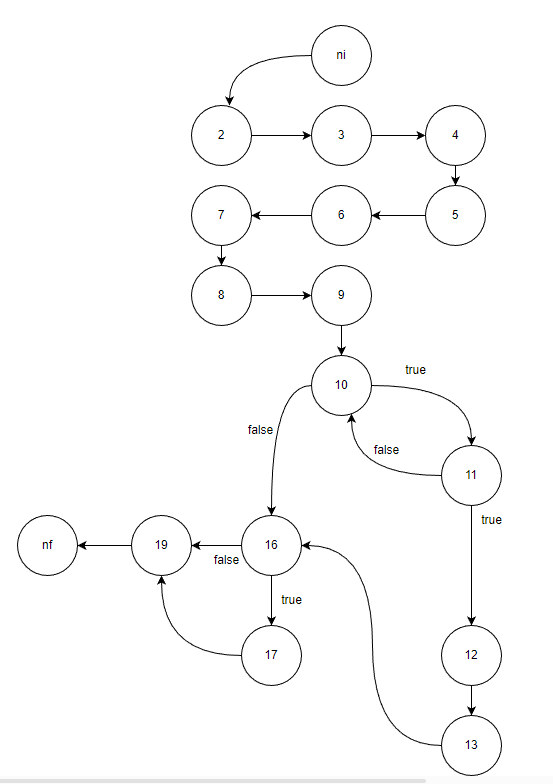
TC8 : (MININT, 0) CE2/CE1 estremi

TC9 : (-1, MAXINT) CE2/CE1 estremi

TC10 : (-10, -10) CE2/CE2

TC11 : (MININT, MININT) CE2/CE2 estremi

Immagine che contiene testo

Descrizione generata automaticamenteTC12 : ( -1, -1) CE2/CE2 estremi

NODE COVERAGE

TC1 = (idAvvisoPresenteInTabella, idUtenteEsistenteNelDatabase (che ha un avviso letto))

TC2 = (idAvvisoPresentaInTabella, idUtenteEsistenteNelDatabase (che non ha un avviso letto))

BRANCH COVERAGE

TC1 = (idAvvisoNonPresenteInTabella, Qualunque idUtente) :: Throws SQLException

TC2 = (idAvvisoPresenteInTabella, idUtenteEsistenteNelDatabase (che ha un avviso letto))

Fino a 11 – 12 – 13 – 16 – 19

TC2 = (idAvvisoPresenteInTabella, idUtenteEsistenteNelDatabase (che non ha un avviso letto))

Fino a 11 – 10 – 11 – 16 – 17 – 19

TC3 = (

13 – 16 – 17 non si può percorrere

**Immagine che contiene testo

Descrizione generata automaticamente**

**CREATO UN UTENTE E RI STORANTE DI TESTING**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **2** | **“Teka”** | **"teka.freitas@example.com"** | **"admin"** | **9** |
| **7** | **“Ece”** | **"ece.hakyemez@example.com"** | **"supervisore"** | **2** |
| **10** | **“Cornelio”** | **“cornelio.fuentes@example.com”** | **“addetto\_sala”** | **7** |
| **idUtente** | **nome** | **Email** | **Ruolo** | **idRistorante** |

Immagine che contiene collana

Descrizione generata automaticamenteTESTING WHITE BOX

NODE COVERAGE

TC1 : (9, “testo”, 2) :: Avviso(MAX(id\_avviso), 2, “testo”, orario di aggiunta, “Teka”)

Branch : 2/8 – 9 – 10 – 11 – 12 – 13 – 9 – 18 – 19 – 20/24 – 24 – 25 – 26

TC2 : (idUtente non esistente, null, 2) :: null

Branch : 2/8 – 9 – 18 – 30

BRANCH COVERAGE : 13

Se si percorre il nodo 12 allora si deve percorrere il nodo 19 sennò la branch non è testabile e viceversa.

Non è possibile che il result set restituisca più di un risultato quindi i percorsi che ritornano al nodo 9 due volte non sono percorribili.

L’arco 25 – 30 è impossibile da verificare poiché il result set restituirà sempre l’avviso appena inserito.

8 - 9 - 10 - 11 - 12 - 13 - 9 - 18 - 30

8 - 9 - 19 - 30

8 - 9 - 19 - 20 - 25 - 30

8 - 9 - 10 - 11 - 9 - 18 - 19 - 20 - 25 - 30

8 - 9 - 19 - 20 - 25 - 26

8 - 9 - 10 - 11 - 9 - 18 - 19 - 20 - 25 - 26

8 - 9 - 10 - 11 - 9 - 10 - 11 - 12 - 13 - 9 - 18 - 19 - 30

8 - 9 - 10 - 11 - 9 - 10 - 11 - 12 - 13 - 9 - 18 - 19 - 20 - 25 - 30

8 - 9 - 10 - 11 - 9 - 10 - 11 - 12 - 13 - 9 - 18 - 19 - 20 - 25 - 26

TC1: (9, null, 2) :: null

2/8 – 9 – 10 – 11 – 12 – 13 – 9 – 18 – 19 – 30

TC2: (9, “testo”, 2) :: Avviso(MAX(id\_avviso), 2, “testo”, orario di aggiunta, “Teka”)

2/8 – 9 – 10 – 11 – 12 – 13 – 9 – 18 – 19 – 20/24 – 25 – 26

TC3: (2, null, idUtente non esistente) :: null

2/8 – 9 – 18 – 30

TC4: (7, “testo”, 10) :: null

2/8 - 9 - 10 - 11 - 9 - 18 - 30

Branch coverage: 4 / 13 = 31%

CONDITION MODIFIED

TC1.1: (2, null, 7)

TC1.2: (9, null, 2)

TC2.1: (2, “testo”, 7)

TC2.2: (9, “testo”, 2)

BLACK BOX

Id\_ristorante

CE1: [idRistorante a cui appartiene idUtente] -----------------------------------------------------------------------VALIDA

CE2: [idRistoranti a cui non appartiene idUtente] ------------------------------------------------------------------VALIDA

CE3: [MININT, -1] -----------------------------------------------------------------------------------------------------------NON VALIDA

Testo

CE1: [String] --------------------------------------------------------------------------------------------------------VALIDA

CE2: [null] -----------------------------------------------------------------------------------------------------------------------NON VALIDA

IdUtente

CE1: [idUtente presente nel db] ---------------------------------------------------------------------------------------- VALIDA

CE2: [idUtente non presente nel db] ---------------------------------------------------------------------------------- VALIDA

CE3: [MININT, -1] -----------------------------------------------------------------------------------------------------------NON VALIDA

SECT - 3 \* 2 \* 3 = 18

TC1: (9, “testo”, 2) :: Avviso(MAX(id\_avviso), 2, “testo”, orario di aggiunta, “Teka”) (CE1, CE1, CE1)

TC2: (9, “testo”, idUtente non presente nel db) :: null (CE1, CE1, CE2)

TC3: (9, “testo”, -10) :: null (CE1, CE1, CE3)

TC4: (9, null, 2) :: null (CE1, CE2, CE1)

TC5: (9, null, idUtente non presente nel db) :: null (CE1, CE2, CE2)

TC6: (9, null, -10) :: null (CE1, CE2, CE3)

TC7: (10, “testo”, 2) :: null (CE2, CE1, CE1)

TC8: (10, “testo”, idUtente non presente nel db) :: null (CE2, CE1, CE2)

TC9: (10, “testo”, -10) :: null (CE2, CE1, CE3)

TC10: (10, null, 2) :: null (CE2, CE2, CE1)

TC11: (10, null, idUtente non presente nel db) :: null (CE2, CE2, CE2)

TC12: (10, null, -10) :: null (CE2, CE2, CE3)

TC13: (-10, “testo”, 2) :: null (CE3, CE1, CE1)

TC14: (-10, “testo”, idUtente non presente nel db) :: null (CE3, CE1, CE2)

TC15: (-10, “testo”, -10) :: null (CE3, CE1, CE3)

TC16: (-10, null, 2) :: null (CE3, CE2, CE1)

TC17: (-10, null, idUtente non presente nel db) :: null (CE3, CE2, CE2)

TC18: (-10, null, -10) :: null (CE3, CE2, CE3)