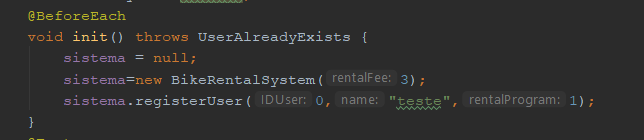
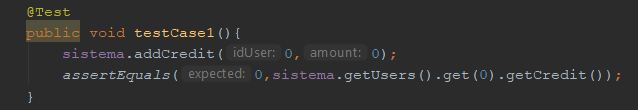
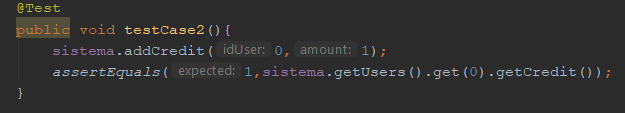
addCreditTest



Este bloco representa as intruções executadas antes de cada teste, é instanciada uma variavél do tipo sistema com o rentalFee=3, depois é adicionado um utilizador com id=0,name = “teste” e rentalProgram = 1.

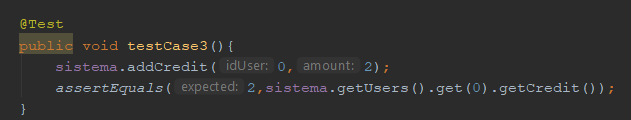


TestCase1: neste teste é adicionado a um utilizador já existente um amount inválido, os resultados esperados são que o método não seja executado e não seja adicionado créditos á conta do utilizador



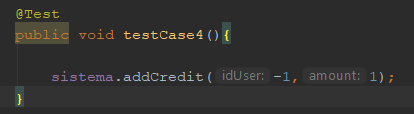
TestCase2: neste teste é adicionado a um utilizador já existente um amount válido = 1, os resultados esperados é que o crédito seja adicionado e que o balanço da conta do utilizador seja 1.

Test Status: Passed



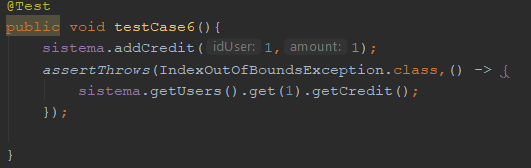
TestCase3: neste teste é adicionado a um utilizador já existente um amount válido = 2, os resultados esperados é que o crédito seja adicionado e que o balanço da conta do utilizador seja 2.

Test Status: Passed



TestCase4: neste teste é enviado para o método addCredit() um idUser inválido, o resultado esperado é que o método não seja executado.

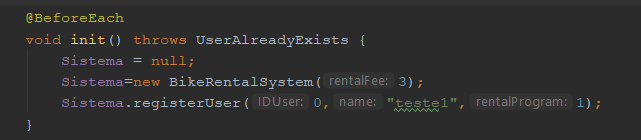
TestCase5: semelhante ao TestCase 2.



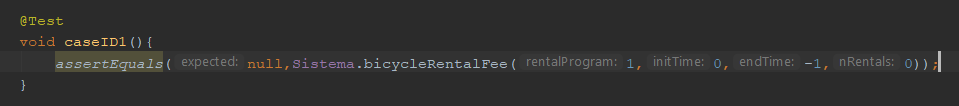
TestCase6: neste teste tentamos adicionar a crédito a um utilizador não existente, o resultado esperado é uma exceção do tipo “IndexOutOfBoundsException”.

Test Status: Passed

BicycleRentalFeeTest

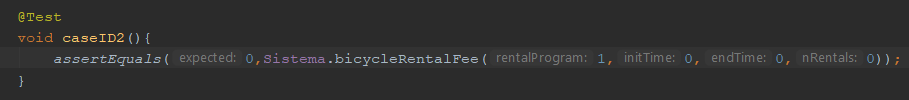


Este bloco representa as intruções executadas antes de cada teste, é instanciada uma variavél do tipo sistema com o rentalFee=3, depois é adicionado um utilizador com id=0,name = “teste1” e rentalProgram = 1.



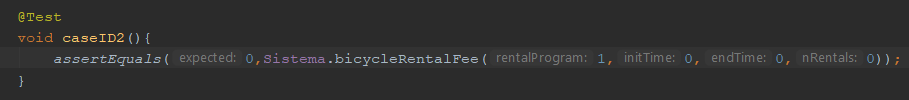
TestCase1: neste teste tentamos enviar para o método bicycleRentalFee() um endTime inválido, o resultado esperado é que o método não seja executado mas na verdade é retornado o valor -3.

Test Status: Failed



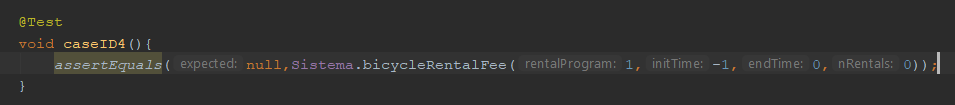
TestCase2: neste teste todas as variáveis enviadas para o método são válidas( rentalProgram = 1, initTime = 0, endTime = 0, nRentals = 0), o resultado esperado é o valor 0.

Test Status: Passed



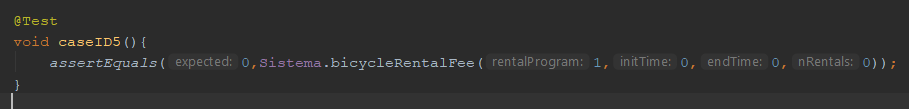
TestCase3: neste teste todas as variáveis enviadas para o método são válidas( rentalProgram = 1, initTime = 0, endTime = 1, nRentals = 0), o resultado esperado é o valor 3.

Test Status: Passed

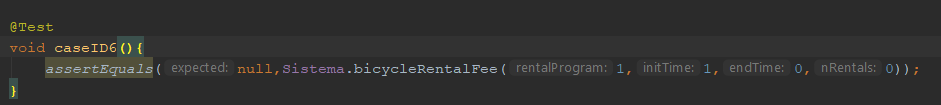


TestCase4: neste teste enviamos para o método bicycleRentalFee() um initTime inválido, o resultado esperado é que o método não seja executado mas na verdade é retornado o valor 3.

Test Status: Failed

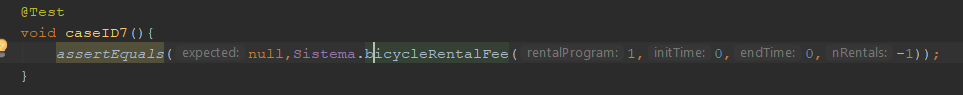


TestCase5: semelhante ao TestCase2.



TestCase6: neste teste é enviado para o método bicycleRentalFee() um initTime superior ao endTime (condição inválida), o resultado esperado é que o método não seja executado mas na verdade é retornado o valor -3.

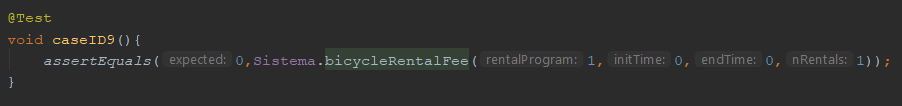
Test Status: Failed



TestCase7: neste teste é enviado para o método bicycleRentalFee() um nRentals inválido = -1, o resultado esperado é que o método não seja executado mas na verdade é retornado o valor 0;

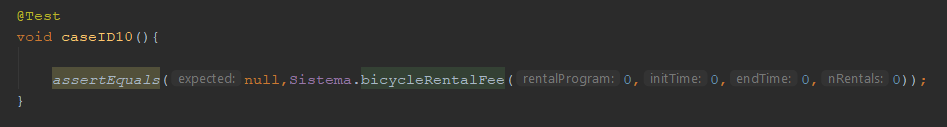
Test Status: Failed

TestCase8: semelhante ao TestCase2.



TestCase9: neste teste todas as variáveis enviadas para o método são válidas( rentalProgram = 1, initTime = 0, endTime = 0, nRentals = 1), o resultado esperado é o valor 0.

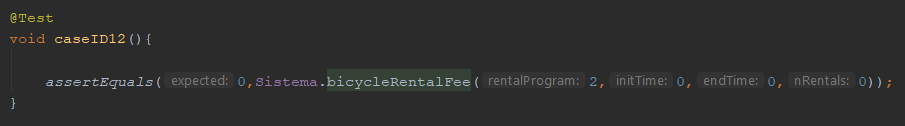
Test Status: Passed



TestCase10: neste teste é enviado para o método bicycleRentalFee() um rentalProgram inválido = 0, o resultado esperado é que o método não seja executado mas na verdade é retornado o valor 0;

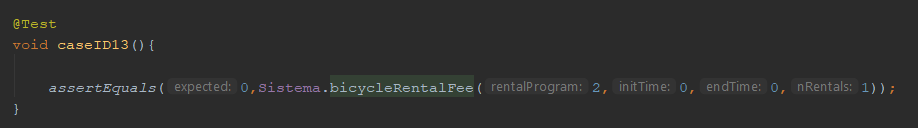
Test Status: Failed

TesteCase11: semelhante ao TestCase2.



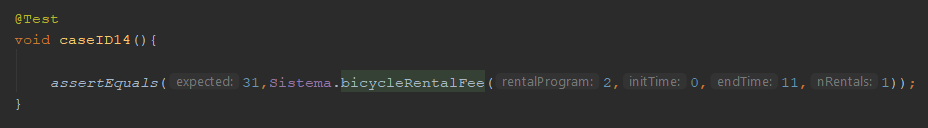
TestCase12:neste teste todas as variáveis enviadas para o método são válidas( rentalProgram = 2, initTime = 0, endTime = 0, nRentals = 0), o resultado esperado é o valor 0.

Test Status: Passed



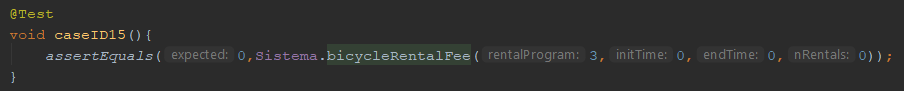
TestCase13: neste teste todas as variáveis enviadas para o método são válidas( rentalProgram = 2, initTime = 0, endTime = 0, nRentals = 1), o resultado esperado é o valor 0.

Test Status: Passed



TestCase14: neste teste todas as variáveis enviadas para o método são válidas( rentalProgram = 2, initTime = 0, endTime = 11, nRentals = 1), o resultado esperado é o valor 31.

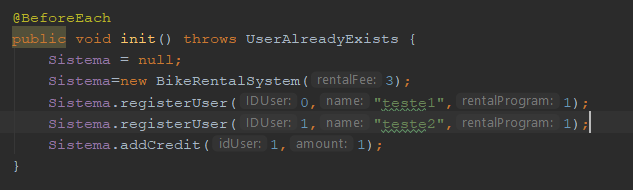
Test Status: Passed



TestCase15: neste teste é enviado para o método bicycleRentalFee() um rentalProgram que é válido mas não existe e o resultado esperado é o valor 0.

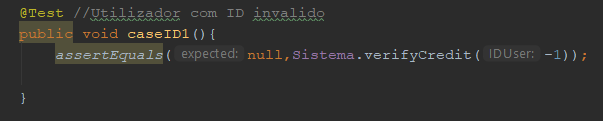
Test Status: Passed

verifyCreditTest



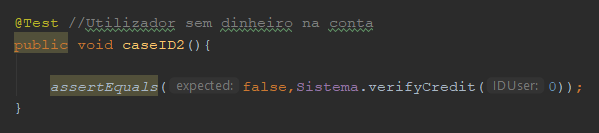
Este bloco representa as intruções executadas antes de cada teste, é instanciada uma variavél do tipo sistema com o rentalFee=3, depois é adicionado um utilizador com id=0,name = “teste1” e rentalProgram = 1 e outro com id=1, name=”teste2” e rentalProgram =1.

Ao utilzador com id = 1 são adicionados créditos na conta enquando que o utilizador com id = 0 tem a conta com 0 créditos.



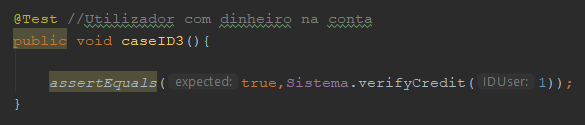
TestCase1: neste teste é enviado para o método verifyCredit() um IDUser inválido = -1, o resultado esperado é que o método não seja executado mas na verdade é retornado False.

Test Status: Failed



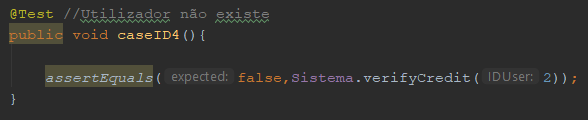
TestCase2: neste teste é enviado para o método verifyCredit() um IDUser de um utilizador que não tem créditos na conta, o resultado esperado é um false.

Test Status: Passed



TestCase3: neste teste é enviado para o método verifyCredit() um IDUser de um utilizador que tem créditos na conta, o resultado esperado é um true.

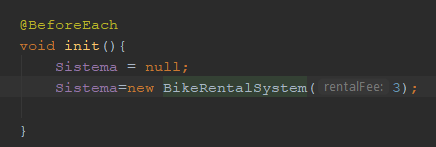
Test Status: Passed



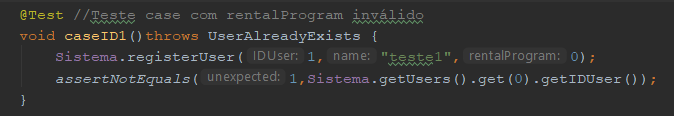
TestCase4: neste teste é enviado para o método verifyCredit() um IDUser de um utilizador que não existe, o resultado esperado é um false.

Test Status: Passed

RegisterUserTest

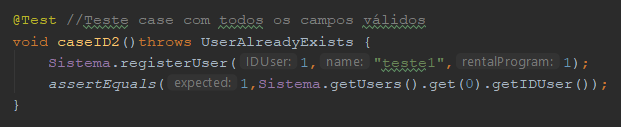


Este bloco representa as intruções executadas antes de cada teste, é instanciada uma variavél do tipo sistema com o rentalFee=3



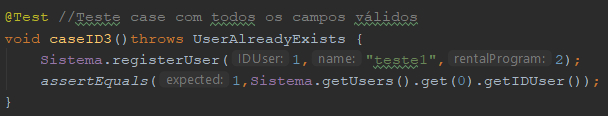
TestCase1: Neste teste é enviado para o método registerUser() um rentalProgram inválido = 0, resultado esperado é que o método não seja executado mas na verdade o utilizador é adicionado ao sistema com um rentalProgram inválido

Test Status: Failed



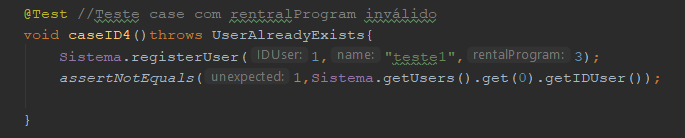
TestCase2: neste teste todas as variáveis enviadas para o método são válidas( IDUser = 1, name=”teste1”,rentalProgam=1), o resultado esperado é que o utilizador seja adicionado ao sistema com sucesso.

Test Results: Passed



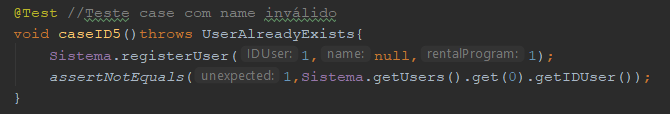
TestCase3: neste teste todas as variáveis enviadas para o método são válidas( IDUser = 1, name=”teste1”,rentalProgam=2), o resultado esperado é que o utilizador seja adicionado ao sistema com sucesso.

Test Results: Passed



TestCase4: Neste teste é enviado para o método registerUser() um rentalProgram inválido = 3,o resultado esperado é que o método não seja executado mas na verdade o utilizador é adicionado ao sistema com um rentalProgram inválido.

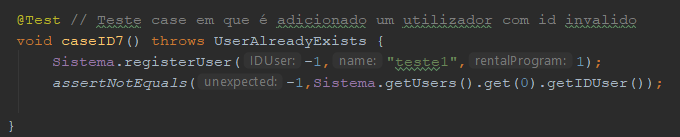
Test Status: Failed



TestCase5: Neste teste é enviado para o método registerUser() um name inválido = null,o resultado esperado é que o método não seja executado mas na verdade o utilizador é adicionado ao sistema com um name inválido.

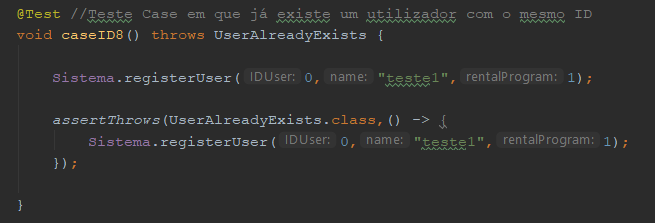
Test Results: Failed

TestCase6: Semelhante ao TestCase2



TestCase7: Neste teste é enviado para o método registerUser() um IDUser inválido = -1,o resultado esperado é que o método não seja executado mas na verdade o utilizador é adicionado ao sistema com um IDUser inválido.

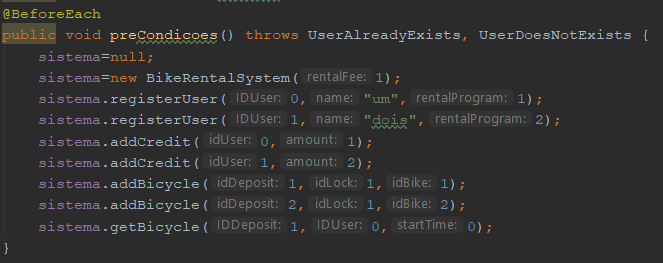
Test Status: Failed



TestCase8: Neste teste tentamos adicionar um utilizador com um IDUser igual ao de um utilizador já registado no sistema, o resultado esperado é um exceção do tipo “UserAlreadyExists”.

Test Status: Passed

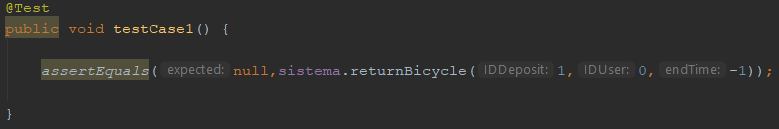
ReturnBicycle



Este bloco representa as intruções executadas antes de cada teste, é instanciada uma variavél do tipo sistema com o rentalFee=3, depois é adicionado um utilizador com id=0,name = “um” e rentalProgram = 1 e outro com id=1, name=”dois” e rentalProgram =2.

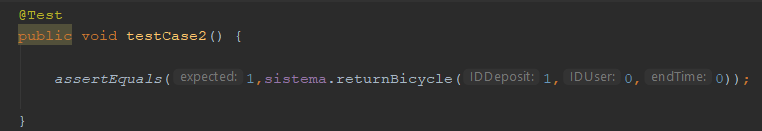
Ao utilzador com id = 1 são adicionados créditos na conta com amount=2 e ao utilizador com id = 0 são adicionados créditos na conta com amount = 1.

De seguida são adicionadas duas bicicletas em depositos diferentes e é feito o get da bicicleta guardada no depósito 1 para o o utilizador com id=0.



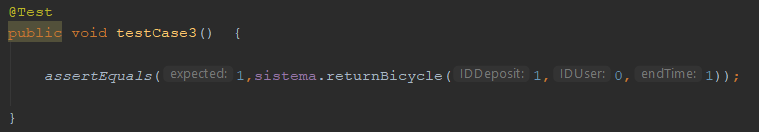
TestCase1: neste teste enviamos para o método returnBicycle() um endTime inválido = -1, o resultado esperado é que o método não seja executado mas na verdade é retornado o valor 1.

Test Status: Failed



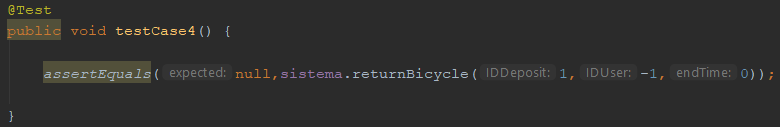
TestCase2: neste teste todas as variáveis enviadas para o método são válidas(IDDeposit= 1, IDUser = 0 ,endTime = 0), o resultado esperado é o valor 1.

Test Status: Passed



TestCase3: neste teste todas as variáveis enviadas para o método são válidas(IDDeposit= 1, IDUser = 0 ,endTime = 1), o resultado esperado é o valor 1.

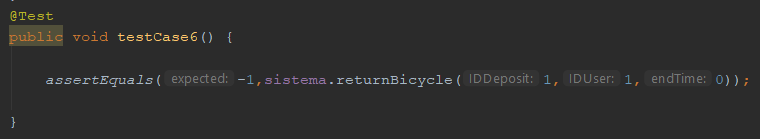
Test Status: Passed



TestCase4: neste teste enviamos para o método returnBicycle() um IDUser inválido = -1, o resultado esperado é que o método não seja executado mas na verdade é retornado o valor -1.

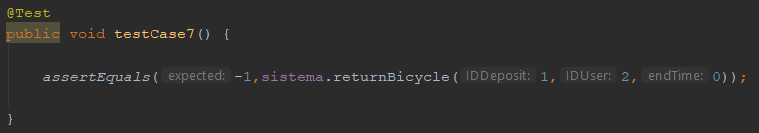
Test Status: Failed

TestCase5: Semelhante ao TestCase2



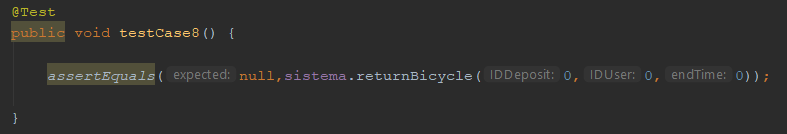
TestCase6: neste teste todas as variáveis enviadas para o método são válidas(IDDeposit= 1, IDUser = 1 ,endTime = 0), o utilizador neste caso não tem uma bicicleta associada, o resultado esperado é o valor -1.

Test Status: Passed



TestCase7: neste teste todas as variáveis enviadas para o método são válidas(IDDeposit= 1, IDUser = 2 ,endTime = 0) mas o utilizador não está registado no sistema, o resultado esperado é valor -1.

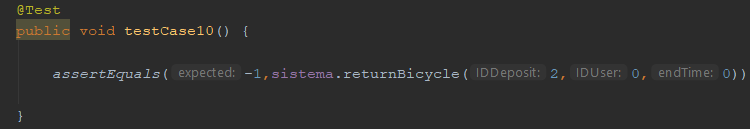
Test Status: Passed



TestCase8: neste teste enviamos para o método returnBicycle() um IDDeposit inválido = 0, o resultado esperado é que o método não seja executado mas na verdade é retornado o valor -1.

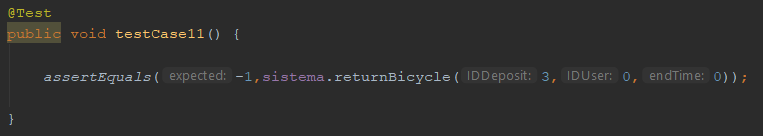
Test Status: Failed

TestCase9: Semelhante ao TestCase2.



TestCase10: neste teste todas as variáveis enviadas para o método são válidas(IDDeposit= 2, IDUser = 0 ,endTime = 0), mas o depósito não tem espaço, o resultado esperado é o valor -1.

Test Status: Passed



TestCase11: neste teste todas as variáveis enviadas para o método são válidas(IDDeposit= 3, IDUser = 0 ,endTime = 0), mas o depósito não, o resultado esperado é o valor -1.

Test Status: Passed

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **TEST CASES** | | | | | | | | |
| **TEST CASE ID** | **TEST CASE** | **PRE-CONDITIONS** | **VARIABLES** | | | **EXPECTED RESULTS** | **ACTUAL RESULTS** | **Test Status** |
| IDDeposit | IDUser | starttime |
| 1 | starttime inválido | -Utilizador,id=1,crédito=1,  semBike -Utilizador,id=2,credito=1,  comBike -Utilizador,id=3,crédito=0,  comBike -Deposito,id=1,comBike -Deposito,id=2,semBike | 1 | 1 | -1 | Não entra no método | 1 | Failed |
| 2 | starttime válido | 1 | 1 | 0 | Diz tempo inicial e return 1 | 1 | Passed |
| 3 | starttime válido | 1 | 1 | 1 | Diz tempo inicial e return 1 | 1 | Passed |
| 4 | IDUser inválido | 1 | 0 | 0 | Não entra no método | Exceptions UserDoesNotExists | Failed |
| 5 | IDUser válido | 1 | 1 | 0 | Diz tempo inicial e return 1 | 1 | Passed |
| 6 | IDUser válido já tem aluguer ativo | 1 | 2 | 0 | return -1 (já tem bike) | 1 | Failed |
| 7 | IDUser válido não tem crédito | 1 | 3 | 0 | return -1 (não tem crédito) | -1 | Passed |
| 8 | IDUser válido não existente | 1 | 4 | 0 | exceção user nao existe | Exceptions UserDoesNotExists | Passed |
| 9 | IDDeposit inválido | 0 | 1 | 0 | Não entra no método | -1 | Failed |
| 10 | IDDeposit válido | 1 | 1 | 0 | Diz tempo inicial e return 1 | 1 | Passed |
| 11 | IDDeposit válido não tem Bike | 2 | 1 | 0 | return -1 (deposito sem bike) | -1 | Passed |
| 12 | IDDeposit válido não existente | 3 | 1 | 0 | return -1 (deposito nao existe) | -1 | Passed |