

NeverEmpty

Generated by Doxygen 1.8.14



# Contents

<b>1</b>	<b>BSPQ18-S5</b>	<b>1</b>
<b>2</b>	<b>Namespace Index</b>	<b>3</b>
2.1	Packages . . . . .	3
<b>3</b>	<b>Hierarchical Index</b>	<b>5</b>
3.1	Class Hierarchy . . . . .	5
<b>4</b>	<b>Class Index</b>	<b>7</b>
4.1	Class List . . . . .	7
<b>5</b>	<b>File Index</b>	<b>11</b>
5.1	File List . . . . .	11
<b>6</b>	<b>Namespace Documentation</b>	<b>13</b>
6.1	Package SPQ . . . . .	13
6.1.1	Detailed Description . . . . .	14
6.2	Package SPQ.controller . . . . .	14
6.2.1	Detailed Description . . . . .	14
6.3	Package SPQ.dao . . . . .	14
6.4	Package SPQ.data . . . . .	15
6.4.1	Detailed Description . . . . .	15
6.5	Package SPQ.dto . . . . .	15
6.5.1	Detailed Description . . . . .	15
6.6	Package SPQ.gateway . . . . .	15
6.7	Package SPQ.gui . . . . .	16
6.7.1	Detailed Description . . . . .	16
6.8	Package SPQ.gui.component . . . . .	16
6.8.1	Detailed Description . . . . .	17
6.9	Package SPQ.main . . . . .	17
6.9.1	Detailed Description . . . . .	17
6.10	Package SPQ.remote . . . . .	17
6.10.1	Detailed Description . . . . .	17

<b>7 Class Documentation</b>	<b>19</b>
7.1 SPQ.data.Account Class Reference . . . . .	19
7.1.1 Detailed Description . . . . .	19
7.1.2 Constructor & Destructor Documentation . . . . .	19
7.1.2.1 Account() [1/2] . . . . .	19
7.1.2.2 Account() [2/2] . . . . .	20
7.1.3 Member Function Documentation . . . . .	20
7.1.3.1 getBalance() . . . . .	20
7.1.3.2 getCardholder() . . . . .	21
7.1.3.3 getCardnumber() . . . . .	21
7.1.3.4 getCvv() . . . . .	21
7.1.3.5 getExpDate() . . . . .	22
7.1.3.6 setBalance() . . . . .	22
7.1.3.7 setCardholder() . . . . .	22
7.1.3.8 setCardnumber() . . . . .	22
7.1.3.9 setCvv() . . . . .	23
7.1.3.10 setExpDate() . . . . .	23
7.2 SPQ.dao.AccountDAO Class Reference . . . . .	23
7.2.1 Detailed Description . . . . .	24
7.2.2 Constructor & Destructor Documentation . . . . .	24
7.2.2.1 AccountDAO() . . . . .	24
7.2.3 Member Function Documentation . . . . .	24
7.2.3.1 pay() . . . . .	24
7.2.3.2 storeAccount() . . . . .	25
7.3 SPQ.ClientTest Class Reference . . . . .	25
7.3.1 Detailed Description . . . . .	26
7.3.2 Member Function Documentation . . . . .	26
7.3.2.1 restoreUser() . . . . .	26
7.3.2.2 setUp() . . . . .	27
7.3.2.3 testGetProducts() . . . . .	27

7.3.2.4	test GetUser()	28
7.3.2.5	test Login()	28
7.3.2.6	test Login Fails()	28
7.3.2.7	test Pay With PayPal()	29
7.3.2.8	test Pay With Visa()	29
7.3.2.9	test Register User Facebook()	30
7.3.2.10	test Register User Google()	30
7.3.2.11	test Register User Never Empty()	31
7.3.2.12	test Set Up Bad Arguments()	31
7.3.2.13	test Update User()	32
7.4	SPQ.CoberturaTest Class Reference	32
7.4.1	Detailed Description	33
7.4.2	Member Function Documentation	33
7.4.2.1	test()	33
7.5	SPQ.DAOMockTest Class Reference	35
7.5.1	Detailed Description	35
7.5.2	Member Function Documentation	35
7.5.2.1	setUp()	35
7.5.2.2	suite()	36
7.5.2.3	test Register User Already Exists()	36
7.5.2.4	test Register User Correctly()	36
7.6	SPQ.DATAProductTest Class Reference	37
7.6.1	Detailed Description	37
7.6.2	Member Function Documentation	37
7.6.2.1	Product DTO Test()	37
7.6.2.2	Product Test()	38
7.6.2.3	setUp()	38
7.7	SPQ.DATAUserTest Class Reference	39
7.7.1	Detailed Description	39
7.7.2	Member Function Documentation	39

7.7.2.1	Alvarotest()	39
7.7.2.2	Cristiantest()	40
7.7.2.3	Ekaitztest()	41
7.7.2.4	Enaratest()	42
7.7.2.5	Jesustest()	43
7.7.2.6	setUp()	43
7.7.2.7	UserDTOTest()	44
7.8	SPQ.gateway.Eroski Class Reference	44
7.8.1	Detailed Description	45
7.8.2	Constructor & Destructor Documentation	45
7.8.2.1	Eroski()	45
7.8.3	Member Function Documentation	45
7.8.3.1	getProducts()	45
7.9	SPQ.EroskiServer Class Reference	46
7.9.1	Detailed Description	46
7.9.2	Member Function Documentation	46
7.9.2.1	main()	46
7.10	SPQ.EroskiService Class Reference	47
7.10.1	Detailed Description	47
7.10.2	Constructor & Destructor Documentation	47
7.10.2.1	EroskiService()	48
7.10.3	Member Function Documentation	48
7.10.3.1	readData()	48
7.10.3.2	run()	49
7.11	SPQ.gateway.Facebook Class Reference	49
7.11.1	Detailed Description	50
7.11.2	Constructor & Destructor Documentation	50
7.11.2.1	Facebook()	50
7.11.3	Member Function Documentation	50
7.11.3.1	register()	50

---

7.12 SPQ.FacebookServer Class Reference . . . . .	51
7.12.1 Detailed Description . . . . .	51
7.12.2 Member Function Documentation . . . . .	51
7.12.2.1 main() . . . . .	51
7.13 SPQ.FacebookService Class Reference . . . . .	52
7.13.1 Detailed Description . . . . .	52
7.13.2 Constructor & Destructor Documentation . . . . .	52
7.13.2.1 FacebookService() . . . . .	53
7.13.3 Member Function Documentation . . . . .	53
7.13.3.1 readData() . . . . .	53
7.13.3.2 run() . . . . .	54
7.14 SPQ.gateway.Google Class Reference . . . . .	54
7.14.1 Detailed Description . . . . .	55
7.14.2 Constructor & Destructor Documentation . . . . .	55
7.14.2.1 Google() . . . . .	55
7.14.3 Member Function Documentation . . . . .	55
7.14.3.1 register() . . . . .	56
7.15 SPQ.GoogleServer Class Reference . . . . .	56
7.15.1 Detailed Description . . . . .	56
7.15.2 Member Function Documentation . . . . .	57
7.15.2.1 main() . . . . .	57
7.16 SPQ.GoogleService Class Reference . . . . .	57
7.16.1 Detailed Description . . . . .	58
7.16.2 Constructor & Destructor Documentation . . . . .	58
7.16.2.1 GoogleService() . . . . .	58
7.16.3 Member Function Documentation . . . . .	58
7.16.3.1 readData() . . . . .	58
7.16.3.2 run() . . . . .	59
7.17 SPQ.dao.IAccountDAO Interface Reference . . . . .	59
7.17.1 Detailed Description . . . . .	60

---

7.17.2 Member Function Documentation . . . . .	60
7.17.2.1 pay() . . . . .	60
7.18 SPQ.gateway.IEroskiGateway Interface Reference . . . . .	60
7.18.1 Detailed Description . . . . .	61
7.18.2 Member Function Documentation . . . . .	61
7.18.2.1 getProducts() . . . . .	61
7.19 SPQ.gateway.IFacebookGateway Interface Reference . . . . .	61
7.19.1 Detailed Description . . . . .	61
7.19.2 Member Function Documentation . . . . .	62
7.19.2.1 register() . . . . .	62
7.20 SPQ.gateway.IGoogleGateway Interface Reference . . . . .	62
7.20.1 Detailed Description . . . . .	62
7.20.2 Member Function Documentation . . . . .	62
7.20.2.1 register() . . . . .	63
7.21 SPQ.remote.INeverEmptyFacade Interface Reference . . . . .	63
7.21.1 Detailed Description . . . . .	64
7.21.2 Member Function Documentation . . . . .	64
7.21.2.1 getProducts() . . . . .	64
7.21.2.2 getUser() . . . . .	64
7.21.2.3 login() . . . . .	65
7.21.2.4 payWithPaypal() . . . . .	65
7.21.2.5 payWithVisa() . . . . .	65
7.21.2.6 registerFacebook() . . . . .	66
7.21.2.7 registerGoogle() . . . . .	66
7.21.2.8 registerNeverEmpty() . . . . .	66
7.21.2.9 updateUser() . . . . .	67
7.22 SPQ.gateway.IPayPalGateway Interface Reference . . . . .	67
7.22.1 Detailed Description . . . . .	67
7.22.2 Member Function Documentation . . . . .	68
7.22.2.1 pay() . . . . .	68

---

7.23 SPQ.dao.IUserDAO Interface Reference . . . . .	68
7.23.1 Detailed Description . . . . .	68
7.23.2 Member Function Documentation . . . . .	68
7.23.2.1 getUser() [1/2] . . . . .	69
7.23.2.2 getUser() [2/2] . . . . .	69
7.23.2.3 pay() . . . . .	69
7.23.2.4 storeUser() . . . . .	69
7.24 SPQ.gateway.IVisaGateway Interface Reference . . . . .	69
7.24.1 Detailed Description . . . . .	70
7.24.2 Member Function Documentation . . . . .	70
7.24.2.1 pay() . . . . .	70
7.25 SPQ.main.Main Class Reference . . . . .	70
7.25.1 Detailed Description . . . . .	71
7.25.2 Member Function Documentation . . . . .	71
7.25.2.1 main() . . . . .	71
7.26 SPQ.controller.NeverEmptyController Class Reference . . . . .	71
7.26.1 Detailed Description . . . . .	72
7.26.2 Constructor & Destructor Documentation . . . . .	72
7.26.2.1 NeverEmptyController() . . . . .	72
7.26.3 Member Function Documentation . . . . .	73
7.26.3.1 getProducts() . . . . .	73
7.26.3.2 getUser() . . . . .	74
7.26.3.3 getUserDTO() . . . . .	75
7.26.3.4 login() . . . . .	75
7.26.3.5 payWithPayPal() . . . . .	76
7.26.3.6 payWithVisa() . . . . .	77
7.26.3.7 registerFacebook() . . . . .	78
7.26.3.8 registerGoogle() . . . . .	79
7.26.3.9 registerNeverEmpty() . . . . .	80
7.26.3.10 setUserDTO() . . . . .	81

7.26.3.11 updateUser() . . . . .	82
7.27 SPQ.remote.NeverEmptyFacade Class Reference . . . . .	83
7.27.1 Detailed Description . . . . .	84
7.27.2 Constructor & Destructor Documentation . . . . .	84
7.27.2.1 NeverEmptyFacade() . . . . .	84
7.27.3 Member Function Documentation . . . . .	85
7.27.3.1 getProducts() . . . . .	85
7.27.3.2 getUser() . . . . .	85
7.27.3.3 login() . . . . .	86
7.27.3.4 payWithPaypal() . . . . .	86
7.27.3.5 payWithVisa() . . . . .	87
7.27.3.6 registerFacebook() . . . . .	88
7.27.3.7 registerGoogle() . . . . .	88
7.27.3.8 registerNeverEmpty() . . . . .	89
7.27.3.9 updateUser() . . . . .	89
7.28 SPQ.NeverEmptyServer Class Reference . . . . .	90
7.28.1 Detailed Description . . . . .	91
7.28.2 Constructor & Destructor Documentation . . . . .	91
7.28.2.1 NeverEmptyServer() . . . . .	91
7.28.3 Member Function Documentation . . . . .	92
7.28.3.1 getProducts() . . . . .	92
7.28.3.2 getUser() . . . . .	92
7.28.3.3 login() . . . . .	93
7.28.3.4 payWithPaypal() . . . . .	94
7.28.3.5 payWithVisa() . . . . .	95
7.28.3.6 registerFacebook() . . . . .	96
7.28.3.7 registerGoogle() . . . . .	97
7.28.3.8 registerNeverEmpty() . . . . .	98
7.28.3.9 updateUser() . . . . .	98
7.29 SPQ.NeverEmptyServerManager Class Reference . . . . .	99

7.29.1 Detailed Description . . . . .	100
7.29.2 Member Function Documentation . . . . .	100
7.29.2.1 main() . . . . .	100
7.30 SPQ.dto.PaymentDTO Class Reference . . . . .	101
7.30.1 Detailed Description . . . . .	102
7.30.2 Constructor & Destructor Documentation . . . . .	102
7.30.2.1 PaymentDTO() [1/6] . . . . .	102
7.30.2.2 PaymentDTO() [2/6] . . . . .	103
7.30.2.3 PaymentDTO() [3/6] . . . . .	103
7.30.2.4 PaymentDTO() [4/6] . . . . .	103
7.30.2.5 PaymentDTO() [5/6] . . . . .	104
7.30.2.6 PaymentDTO() [6/6] . . . . .	104
7.30.3 Member Function Documentation . . . . .	104
7.30.3.1 getCardholder() [1/3] . . . . .	104
7.30.3.2 getCardholder() [2/3] . . . . .	104
7.30.3.3 getCardholder() [3/3] . . . . .	105
7.30.3.4 getCardnumber() [1/3] . . . . .	105
7.30.3.5 getCardnumber() [2/3] . . . . .	105
7.30.3.6 getCardnumber() [3/3] . . . . .	105
7.30.3.7 getCvv() [1/3] . . . . .	106
7.30.3.8 getCvv() [2/3] . . . . .	106
7.30.3.9 getCvv() [3/3] . . . . .	106
7.30.3.10 getExpDate() [1/3] . . . . .	106
7.30.3.11 getExpDate() [2/3] . . . . .	107
7.30.3.12 getExpDate() [3/3] . . . . .	107
7.30.3.13 getPassword() [1/3] . . . . .	107
7.30.3.14 getPassword() [2/3] . . . . .	107
7.30.3.15 getPassword() [3/3] . . . . .	108
7.30.3.16 getTotal() [1/3] . . . . .	108
7.30.3.17 getTotal() [2/3] . . . . .	108

7.30.3.18 getTotal() [3/3] . . . . .	108
7.30.3.19 getUsername() [1/3] . . . . .	109
7.30.3.20 getUsername() [2/3] . . . . .	109
7.30.3.21 getUsername() [3/3] . . . . .	109
7.30.3.22 setCardholder() [1/3] . . . . .	109
7.30.3.23 setCardholder() [2/3] . . . . .	110
7.30.3.24 setCardholder() [3/3] . . . . .	110
7.30.3.25 setCardnumber() [1/3] . . . . .	110
7.30.3.26 setCardnumber() [2/3] . . . . .	110
7.30.3.27 setCardnumber() [3/3] . . . . .	110
7.30.3.28 setCvv() [1/3] . . . . .	111
7.30.3.29 setCvv() [2/3] . . . . .	111
7.30.3.30 setCvv() [3/3] . . . . .	111
7.30.3.31 setExpDate() [1/3] . . . . .	111
7.30.3.32 setExpDate() [2/3] . . . . .	111
7.30.3.33 setExpDate() [3/3] . . . . .	112
7.30.3.34 setPassword() [1/3] . . . . .	112
7.30.3.35 setPassword() [2/3] . . . . .	112
7.30.3.36 setPassword() [3/3] . . . . .	112
7.30.3.37 setTotal() [1/3] . . . . .	112
7.30.3.38 setTotal() [2/3] . . . . .	113
7.30.3.39 setTotal() [3/3] . . . . .	113
7.30.3.40 setUsername() [1/3] . . . . .	113
7.30.3.41 setUsername() [2/3] . . . . .	113
7.30.3.42 setUsername() [3/3] . . . . .	113
7.31 SPQ.gateway.PayPal Class Reference . . . . .	114
7.31.1 Detailed Description . . . . .	114
7.31.2 Constructor & Destructor Documentation . . . . .	114
7.31.2.1 PayPal() . . . . .	115
7.31.3 Member Function Documentation . . . . .	115

---

7.31.3.1	pay()	115
7.32	SPQ.PaypalServer Class Reference	115
7.32.1	Detailed Description	116
7.32.2	Member Function Documentation	116
7.32.2.1	main()	116
7.33	SPQ.PaypalService Class Reference	116
7.33.1	Detailed Description	117
7.33.2	Constructor & Destructor Documentation	117
7.33.2.1	PaypalService()	117
7.33.3	Member Function Documentation	117
7.33.3.1	readData()	118
7.33.3.2	run()	118
7.34	SPQ.data.Product Class Reference	119
7.34.1	Detailed Description	120
7.34.2	Constructor & Destructor Documentation	120
7.34.2.1	Product() [1/3]	120
7.34.2.2	Product() [2/3]	120
7.34.2.3	Product() [3/3]	120
7.34.3	Member Function Documentation	121
7.34.3.1	getName() [1/2]	121
7.34.3.2	getName() [2/2]	121
7.34.3.3	getPrice() [1/2]	121
7.34.3.4	getPrice() [2/2]	122
7.34.3.5	getQuantity()	122
7.34.3.6	getSale() [1/2]	122
7.34.3.7	getSale() [2/2]	123
7.34.3.8	setName() [1/2]	123
7.34.3.9	setName() [2/2]	123
7.34.3.10	setPrice() [1/2]	124
7.34.3.11	setPrice() [2/2]	124

7.34.3.12 setQuantity() . . . . .	124
7.34.3.13 setSale() [1/2] . . . . .	125
7.34.3.14 setSale() [2/2] . . . . .	125
7.35 SPQ.dao.ProductDAO Class Reference . . . . .	125
7.35.1 Detailed Description . . . . .	126
7.35.2 Constructor & Destructor Documentation . . . . .	126
7.35.2.1 ProductDAO() . . . . .	126
7.35.3 Member Function Documentation . . . . .	126
7.35.3.1 deleteProduct() . . . . .	126
7.35.3.2 getProducts() . . . . .	126
7.35.3.3 storeProduct() . . . . .	127
7.36 SPQ.dto.ProductDTO Class Reference . . . . .	127
7.36.1 Detailed Description . . . . .	128
7.36.2 Constructor & Destructor Documentation . . . . .	128
7.36.2.1 ProductDTO() [1/2] . . . . .	128
7.36.2.2 ProductDTO() [2/2] . . . . .	128
7.36.3 Member Function Documentation . . . . .	128
7.36.3.1 getProductList() [1/2] . . . . .	129
7.36.3.2 getProductList() [2/2] . . . . .	129
7.36.3.3 setProductList() [1/2] . . . . .	129
7.36.3.4 setProductList() [2/2] . . . . .	129
7.37 SPQ.gui.component.ProductLabel Class Reference . . . . .	130
7.37.1 Detailed Description . . . . .	131
7.37.2 Constructor & Destructor Documentation . . . . .	131
7.37.2.1 ProductLabel() [1/2] . . . . .	131
7.37.2.2 ProductLabel() [2/2] . . . . .	131
7.37.3 Member Function Documentation . . . . .	131
7.37.3.1 actionPerformed() . . . . .	131
7.37.3.2 getbAdd() . . . . .	132
7.37.3.3 getbSub() . . . . .	132

7.37.3.4 <code>getPrice()</code>	132
7.37.3.5 <code>getproductName()</code>	132
7.37.3.6 <code>getQuantity()</code>	133
7.37.3.7 <code>getSale()</code>	133
7.37.3.8 <code>setbAdd()</code>	133
7.37.3.9 <code>setbSub()</code>	134
7.37.3.10 <code>setPrice()</code>	134
7.37.3.11 <code>setproductName()</code>	134
7.37.3.12 <code>setQuantity()</code>	134
7.37.3.13 <code>setSale()</code>	134
7.38 <code>SPQ.remote.RMIServiceLocator</code> Class Reference	135
7.38.1 Detailed Description	135
7.38.2 Member Function Documentation	135
7.38.2.1 <code>getNeverEmptyServer()</code>	135
7.38.2.2 <code>setService()</code>	136
7.39 <code>SPQ.RMITest</code> Class Reference	137
7.39.1 Detailed Description	137
7.39.2 Member Function Documentation	138
7.39.2.1 <code>deleteDatabase()</code>	138
7.39.2.2 <code>loginExistingUserTest()</code>	138
7.39.2.3 <code>loginNewUserTest()</code>	138
7.39.2.4 <code>registerFacebookExistingUserTest()</code>	139
7.39.2.5 <code>registerGoogleExistingUserTest()</code>	139
7.39.2.6 <code>registerUserFacebookTest()</code>	139
7.39.2.7 <code>registerUserGoogleTest()</code>	140
7.39.2.8 <code>setUp()</code>	140
7.39.2.9 <code>setUpClient()</code>	140
7.39.2.10 <code>suite()</code>	140
7.39.2.11 <code>tearDown()</code>	141
7.39.2.12 <code>testGetProducts()</code>	141

7.39.2.13 testPayWithPayPal() . . . . .	141
7.39.2.14 testPayWithVisa() . . . . .	142
7.40 SPQ.data.User Class Reference . . . . .	142
7.40.1 Detailed Description . . . . .	143
7.40.2 Constructor & Destructor Documentation . . . . .	143
7.40.2.1 User() [1/5] . . . . .	144
7.40.2.2 User() [2/5] . . . . .	144
7.40.2.3 User() [3/5] . . . . .	144
7.40.2.4 User() [4/5] . . . . .	144
7.40.2.5 User() [5/5] . . . . .	145
7.40.3 Member Function Documentation . . . . .	146
7.40.3.1 getAddress() . . . . .	146
7.40.3.2 getBalance() . . . . .	146
7.40.3.3 getCardholder() . . . . .	147
7.40.3.4 getCardNumber() . . . . .	147
7.40.3.5 getEmail() [1/4] . . . . .	148
7.40.3.6 getEmail() [2/4] . . . . .	148
7.40.3.7 getEmail() [3/4] . . . . .	148
7.40.3.8 getEmail() [4/4] . . . . .	148
7.40.3.9 getPassword() [1/4] . . . . .	149
7.40.3.10 getPassword() [2/4] . . . . .	149
7.40.3.11 getPassword() [3/4] . . . . .	149
7.40.3.12 getPassword() [4/4] . . . . .	149
7.40.3.13 getPayPalEmail() . . . . .	150
7.40.3.14 getPayPalPassword() . . . . .	150
7.40.3.15 getRegisterMethod() . . . . .	151
7.40.3.16 getUsername() . . . . .	151
7.40.3.17 setAddress() . . . . .	152
7.40.3.18 setBalance() . . . . .	152
7.40.3.19 setCardholder() . . . . .	152

7.40.3.20 setCardNumber() . . . . .	153
7.40.3.21 setEmail() [1/4] . . . . .	153
7.40.3.22 setEmail() [2/4] . . . . .	153
7.40.3.23 setEmail() [3/4] . . . . .	154
7.40.3.24 setEmail() [4/4] . . . . .	154
7.40.3.25 setPassword() [1/4] . . . . .	154
7.40.3.26 setPassword() [2/4] . . . . .	154
7.40.3.27 setPassword() [3/4] . . . . .	155
7.40.3.28 setPassword() [4/4] . . . . .	155
7.40.3.29 setPayPalEmail() . . . . .	155
7.40.3.30 setPayPalPassword() . . . . .	156
7.40.3.31 setRegisterMethod() . . . . .	156
7.40.3.32 setUsername() . . . . .	157
7.40.3.33 updateUser() . . . . .	157
7.41 SPQ.dao.UserDAO Class Reference . . . . .	158
7.41.1 Detailed Description . . . . .	159
7.41.2 Constructor & Destructor Documentation . . . . .	159
7.41.2.1 UserDAO() [1/4] . . . . .	159
7.41.2.2 UserDAO() [2/4] . . . . .	160
7.41.2.3 UserDAO() [3/4] . . . . .	160
7.41.2.4 UserDAO() [4/4] . . . . .	160
7.41.3 Member Function Documentation . . . . .	160
7.41.3.1 getUser() [1/3] . . . . .	160
7.41.3.2 getUser() [2/3] . . . . .	161
7.41.3.3 getUser() [3/3] . . . . .	161
7.41.3.4 main() . . . . .	161
7.41.3.5 pay() . . . . .	162
7.41.3.6 storeUser() [1/2] . . . . .	162
7.41.3.7 storeUser() [2/2] . . . . .	163
7.41.3.8 updateUser() . . . . .	163

7.42 SPQ.dto.UserDTO Class Reference . . . . .	164
7.42.1 Detailed Description . . . . .	165
7.42.2 Constructor & Destructor Documentation . . . . .	165
7.42.2.1 UserDTO() [1/11] . . . . .	166
7.42.2.2 UserDTO() [2/11] . . . . .	166
7.42.2.3 UserDTO() [3/11] . . . . .	167
7.42.2.4 UserDTO() [4/11] . . . . .	167
7.42.2.5 UserDTO() [5/11] . . . . .	168
7.42.2.6 UserDTO() [6/11] . . . . .	168
7.42.2.7 UserDTO() [7/11] . . . . .	168
7.42.2.8 UserDTO() [8/11] . . . . .	169
7.42.2.9 UserDTO() [9/11] . . . . .	169
7.42.2.10 UserDTO() [10/11] . . . . .	170
7.42.2.11 UserDTO() [11/11] . . . . .	170
7.42.3 Member Function Documentation . . . . .	171
7.42.3.1 getAddress() . . . . .	171
7.42.3.2 getCardholder() . . . . .	172
7.42.3.3 getCardNumber() . . . . .	172
7.42.3.4 getEmail() [1/3] . . . . .	173
7.42.3.5 getEmail() [2/3] . . . . .	173
7.42.3.6 getEmail() [3/3] . . . . .	174
7.42.3.7 getPassword() [1/3] . . . . .	174
7.42.3.8 getPassword() [2/3] . . . . .	174
7.42.3.9 getPassword() [3/3] . . . . .	175
7.42.3.10 getPayPalEmail() . . . . .	175
7.42.3.11 getPayPalPassword() . . . . .	176
7.42.3.12 getRegisterMethod() . . . . .	176
7.42.3.13 getUsername() . . . . .	177
7.42.3.14 setAddress() . . . . .	177
7.42.3.15 setCardholder() . . . . .	177

7.42.3.16 setCardNumber() . . . . .	178
7.42.3.17 setEmail() [1/3] . . . . .	178
7.42.3.18 setEmail() [2/3] . . . . .	178
7.42.3.19 setEmail() [3/3] . . . . .	178
7.42.3.20 setPassword() [1/3] . . . . .	178
7.42.3.21 setPassword() [2/3] . . . . .	179
7.42.3.22 setPassword() [3/3] . . . . .	179
7.42.3.23 setPayPalEmail() . . . . .	179
7.42.3.24 setPayPalPassword() . . . . .	179
7.42.3.25 setRegisterMethod() . . . . .	179
7.42.3.26 setUsername() . . . . .	180
7.43 SPQ.Utilities Class Reference . . . . .	180
7.43.1 Detailed Description . . . . .	180
7.43.2 Member Function Documentation . . . . .	180
7.43.2.1 getImageFromResources() . . . . .	180
7.44 SPQ.gui.VentanaInicio Class Reference . . . . .	181
7.44.1 Detailed Description . . . . .	182
7.44.2 Constructor & Destructor Documentation . . . . .	182
7.44.2.1 VentanaInicio() . . . . .	182
7.44.3 Member Function Documentation . . . . .	183
7.44.3.1 iniciarAplicacion() . . . . .	183
7.44.3.2 main() . . . . .	184
7.45 SPQ.gui.Ventana MetodoPago Class Reference . . . . .	184
7.45.1 Detailed Description . . . . .	185
7.45.2 Constructor & Destructor Documentation . . . . .	185
7.45.2.1 Ventana MetodoPago() . . . . .	186
7.45.3 Member Function Documentation . . . . .	186
7.45.3.1 actionPerformed() . . . . .	186
7.45.4 Member Data Documentation . . . . .	186
7.45.4.1 vt . . . . .	186

7.46 SPQ.gui.VentanaPago Class Reference . . . . .	187
7.46.1 Detailed Description . . . . .	188
7.46.2 Constructor & Destructor Documentation . . . . .	188
7.46.2.1 VentanaPago() . . . . .	188
7.46.3 Member Function Documentation . . . . .	189
7.46.3.1 actionPerformed() . . . . .	189
7.46.3.2 calculateProgress() [1/4] . . . . .	189
7.46.3.3 calculateProgress() [2/4] . . . . .	190
7.46.3.4 calculateProgress() [3/4] . . . . .	190
7.46.3.5 calculateProgress() [4/4] . . . . .	190
7.46.4 Member Data Documentation . . . . .	191
7.46.4.1 vmp . . . . .	191
7.47 SPQ.gui.VentanaPerfil Class Reference . . . . .	191
7.47.1 Detailed Description . . . . .	192
7.47.2 Constructor & Destructor Documentation . . . . .	192
7.47.2.1 VentanaPerfil() . . . . .	192
7.47.3 Member Function Documentation . . . . .	193
7.47.3.1 actionPerformed() . . . . .	193
7.47.3.2 passwordModified() . . . . .	194
7.48 SPQ.gui.VentanaPrincipal Class Reference . . . . .	194
7.48.1 Detailed Description . . . . .	195
7.48.2 Constructor & Destructor Documentation . . . . .	196
7.48.2.1 VentanaPrincipal() . . . . .	196
7.48.3 Member Function Documentation . . . . .	196
7.48.3.1 actionPerformed() . . . . .	196
7.48.3.2 main() . . . . .	197
7.48.4 Member Data Documentation . . . . .	197
7.48.4.1 pCatalogue . . . . .	197
7.49 SPQ.gui.VentanaRegistro Class Reference . . . . .	197
7.49.1 Detailed Description . . . . .	198

---

7.49.2 Constructor & Destructor Documentation . . . . .	198
7.49.2.1 VentanaRegistro() . . . . .	199
7.49.3 Member Function Documentation . . . . .	199
7.49.3.1 ComprobacionSignup() . . . . .	199
7.49.3.2 finalizarRegistro() . . . . .	200
7.49.3.3 main() . . . . .	200
7.50 SPQ.gui.VentanaSeleccionarRegistro Class Reference . . . . .	201
7.50.1 Detailed Description . . . . .	201
7.50.2 Constructor & Destructor Documentation . . . . .	202
7.50.2.1 VentanaSeleccionarRegistro() . . . . .	202
7.50.3 Member Function Documentation . . . . .	202
7.50.3.1 actionPerformed() . . . . .	202
7.51 SPQ.gui.VentanaTicket Class Reference . . . . .	203
7.51.1 Detailed Description . . . . .	204
7.51.2 Constructor & Destructor Documentation . . . . .	204
7.51.2.1 VentanaTicket() . . . . .	204
7.51.3 Member Function Documentation . . . . .	204
7.51.3.1 actionPerformed() . . . . .	205
7.51.3.2 main() . . . . .	205
7.51.4 Member Data Documentation . . . . .	205
7.51.4.1 vp . . . . .	205
7.52 SPQ.gateway.Visa Class Reference . . . . .	206
7.52.1 Detailed Description . . . . .	206
7.52.2 Constructor & Destructor Documentation . . . . .	206
7.52.2.1 Visa() . . . . .	207
7.52.3 Member Function Documentation . . . . .	207
7.52.3.1 pay() . . . . .	207
7.53 SPQ.VisaServer Class Reference . . . . .	207
7.53.1 Detailed Description . . . . .	208
7.53.2 Member Function Documentation . . . . .	208
7.53.2.1 main() . . . . .	208
7.54 SPQ.VisaService Class Reference . . . . .	208
7.54.1 Detailed Description . . . . .	209
7.54.2 Constructor & Destructor Documentation . . . . .	209
7.54.2.1 VisaService() . . . . .	209
7.54.3 Member Function Documentation . . . . .	209
7.54.3.1 readData() . . . . .	210
7.54.3.2 run() . . . . .	210

<b>8 File Documentation</b>	<b>213</b>
8.1 EroskiService/src/main/java/SPQ/dao/ProductDAO.java File Reference . . . . .	213
8.2 EroskiService/src/main/java/SPQ/data/Product.java File Reference . . . . .	213
8.3 NeverEmptyServer/src/main/java/SPQ/data/Product.java File Reference . . . . .	213
8.4 EroskiService/src/main/java/SPQ/dto/ProductDTO.java File Reference . . . . .	214
8.5 NeverEmptyServer/src/main/java/SPQ/dto/ProductDTO.java File Reference . . . . .	214
8.6 EroskiService/src/main/java/SPQ/EroskiServer.java File Reference . . . . .	214
8.7 EroskiService/src/main/java/SPQ/EroskiService.java File Reference . . . . .	214
8.8 FacebookService/src/main/java/SPQ/dao/IUserDAO.java File Reference . . . . .	215
8.9 GoogleService/src/main/java/SPQ/dao/IUserDAO.java File Reference . . . . .	215
8.10 PaypalService/src/main/java/SPQ/dao/IUserDAO.java File Reference . . . . .	215
8.11 FacebookService/src/main/java/SPQ/dao/UserDAO.java File Reference . . . . .	215
8.12 GoogleService/src/main/java/SPQ/dao/UserDAO.java File Reference . . . . .	216
8.13 NeverEmptyServer/src/main/java/SPQ/dao/UserDAO.java File Reference . . . . .	216
8.14 PaypalService/src/main/java/SPQ/dao/UserDAO.java File Reference . . . . .	216
8.15 FacebookService/src/main/java/SPQ/data/User.java File Reference . . . . .	216
8.16 GoogleService/src/main/java/SPQ/data/User.java File Reference . . . . .	217
8.17 NeverEmptyServer/src/main/java/SPQ/data/User.java File Reference . . . . .	217
8.18 PaypalService/src/main/java/SPQ/data/User.java File Reference . . . . .	217
8.19 FacebookService/src/main/java/SPQ/dto/UserDTO.java File Reference . . . . .	217
8.20 GoogleService/src/main/java/SPQ/dto/UserDTO.java File Reference . . . . .	218
8.21 NeverEmptyServer/src/main/java/SPQ/dto/UserDTO.java File Reference . . . . .	218
8.22 FacebookService/src/main/java/SPQ/FacebookServer.java File Reference . . . . .	218
8.23 FacebookService/src/main/java/SPQ/FacebookService.java File Reference . . . . .	218
8.24 GoogleService/src/main/java/SPQ/GoogleServer.java File Reference . . . . .	219
8.25 GoogleService/src/main/java/SPQ/GoogleService.java File Reference . . . . .	219
8.26 NeverEmptyClient/src/main/java/SPQ/controller/NeverEmptyController.java File Reference . . . . .	219
8.27 NeverEmptyClient/src/main/java/SPQ/gui/component/ProductLabel.java File Reference . . . . .	219
8.28 NeverEmptyClient/src/main/java/SPQ/gui/VentanaInicio.java File Reference . . . . .	220
8.29 NeverEmptyClient/src/main/java/SPQ/gui/Ventana MetodoPago.java File Reference . . . . .	220

8.30	NeverEmptyClient/src/main/java/SPQ/gui/VentanaPago.java File Reference . . . . .	220
8.31	NeverEmptyClient/src/main/java/SPQ/gui/VentanaPerfil.java File Reference . . . . .	221
8.32	NeverEmptyClient/src/main/java/SPQ/gui/VentanaPrincipal.java File Reference . . . . .	221
8.33	NeverEmptyClient/src/main/java/SPQ/gui/VentanaRegistro.java File Reference . . . . .	221
8.34	NeverEmptyClient/src/main/java/SPQ/gui/VentanaSeleccionarRegistro.java File Reference . . . . .	222
8.35	NeverEmptyClient/src/main/java/SPQ/gui/VentanaTicket.java File Reference . . . . .	222
8.36	NeverEmptyClient/src/main/java/SPQ/main/Main.java File Reference . . . . .	222
8.37	NeverEmptyClient/src/main/java/SPQ/remote/RMIServiceLocator.java File Reference . . . . .	222
8.38	NeverEmptyClient/src/main/java/SPQ/Utilities.java File Reference . . . . .	223
8.39	NeverEmptyClient/src/test/java/SPQ/ClientTest.java File Reference . . . . .	223
8.40	NeverEmptyServer/src/main/java/SPQ/dto/PaymentDTO.java File Reference . . . . .	223
8.41	PaypalService/src/main/java/SPQ/dto/PaymentDTO.java File Reference . . . . .	224
8.42	VisaService/src/main/java/SPQ/dto/PaymentDTO.java File Reference . . . . .	224
8.43	NeverEmptyServer/src/main/java/SPQ/gateway/Eroski.java File Reference . . . . .	224
8.44	NeverEmptyServer/src/main/java/SPQ/gateway/Facebook.java File Reference . . . . .	224
8.45	NeverEmptyServer/src/main/java/SPQ/gateway/Google.java File Reference . . . . .	225
8.46	NeverEmptyServer/src/main/java/SPQ/gateway/IEroskiGateway.java File Reference . . . . .	225
8.47	NeverEmptyServer/src/main/java/SPQ/gateway/IFacebookGateway.java File Reference . . . . .	225
8.48	NeverEmptyServer/src/main/java/SPQ/gateway/IGoogleGateway.java File Reference . . . . .	225
8.49	NeverEmptyServer/src/main/java/SPQ/gateway/IPayPalGateway.java File Reference . . . . .	226
8.50	NeverEmptyServer/src/main/java/SPQ/gateway/IVisaGateway.java File Reference . . . . .	226
8.51	NeverEmptyServer/src/main/java/SPQ/gateway/PayPal.java File Reference . . . . .	226
8.52	NeverEmptyServer/src/main/java/SPQ/gateway/Visa.java File Reference . . . . .	226
8.53	NeverEmptyServer/src/main/java/SPQ/NeverEmptyServer.java File Reference . . . . .	227
8.54	NeverEmptyServer/src/main/java/SPQ/NeverEmptyServerManager.java File Reference . . . . .	227
8.55	NeverEmptyServer/src/main/java/SPQ/remote/INeverEmptyFacade.java File Reference . . . . .	227
8.56	NeverEmptyServer/src/main/java/SPQ/remote/NeverEmptyFacade.java File Reference . . . . .	227
8.57	NeverEmptyServer/src/test/java/SPQ/CoberturaTest.java File Reference . . . . .	228
8.58	NeverEmptyServer/src/test/java/SPQ/DAOMockTest.java File Reference . . . . .	228
8.59	NeverEmptyServer/src/test/java/SPQ/DATAProductTest.java File Reference . . . . .	228
8.60	NeverEmptyServer/src/test/java/SPQ/DATAUserTest.java File Reference . . . . .	228
8.61	NeverEmptyServer/src/test/java/SPQ/RMITest.java File Reference . . . . .	229
8.62	PaypalService/src/main/java/SPQ/PaypalServer.java File Reference . . . . .	229
8.63	PaypalService/src/main/java/SPQ/PaypalService.java File Reference . . . . .	229
8.64	README.md File Reference . . . . .	229
8.65	VisaService/src/main/java/SPQ/dao/AccountDAO.java File Reference . . . . .	229
8.66	VisaService/src/main/java/SPQ/dao/IAccountDAO.java File Reference . . . . .	230
8.67	VisaService/src/main/java/SPQ/data/Account.java File Reference . . . . .	230
8.68	VisaService/src/main/java/SPQ/VisaServer.java File Reference . . . . .	230
8.69	VisaService/src/main/java/SPQ/VisaService.java File Reference . . . . .	230



# **Chapter 1**

## **BSPQ18-S5**

Repository for Team BSPQ18-S5

### **DESCRIPCION:**

Nuestro proyecto consiste en una compra de comida online. Esta estructura como un proyecto de maven Cliente-Servidor.

La parte cliente la delimita la carpeta: NeverEmptyClient.

La parte servidora la delimitan las carpetas: NeverEmptyServer, EroskiService, FacebookService, GoogleService, PaypalService y VisaService.

### **PARA EJECUTAR EL PROYECTO:**

1º Para compilar: mvn clean compile

2º

2.1) Para ejecutar el cliente: mvn exec:java -Pclient

2.2) Para ejecutar el servidor: mvn exec:java -Pserver

3º Para los test: mvn test

Este comando generará el report Contiperf

4º Para generar el report Cobertura: mvn cobertura:cobertura

Para consultar el porcentaje testeado: mvn cobertura:check



# Chapter 2

## Namespace Index

### 2.1 Packages

Here are the packages with brief descriptions (if available):

<a href="#">SPQ</a>	Client root package . . . . .	13
<a href="#">SPQ.controller</a>	This package contains the classes that act as intermediaries between the graphic user interface and the classes that speak with the server . . . . .	14
<a href="#">SPQ.dao</a>		14
<a href="#">SPQ.data</a>		15
<a href="#">SPQ.dto</a>		15
<a href="#">SPQ.gateway</a>		15
<a href="#">SPQ.gui</a>	The <a href="#">SPQ.gui</a> package contains the classes responsible for displaying the graphic user interface of the application . . . . .	16
<a href="#">SPQ.gui.component</a>	The <a href="#">SPQ.gui.component</a> consist of GUI tiny components that are used by other GUI bigger components as windows . . . . .	16
<a href="#">SPQ.main</a>	The <a href="#">SPQ.main</a> contains the main class . . . . .	17
<a href="#">SPQ.remote</a>	The <a href="#">SPQ.remote</a> contains the RMI service locator . . . . .	17



# Chapter 3

## Hierarchical Index

### 3.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

SPQ.data.Account . . . . .	19
SPQ.ClientTest . . . . .	25
SPQ.CoberturaTest . . . . .	32
SPQ.DAOMockTest . . . . .	35
SPQ.DATAProductTest . . . . .	37
SPQ.DATAUserTest . . . . .	39
SPQ.EroskiServer . . . . .	46
SPQ.FacebookServer . . . . .	51
SPQ.GoogleServer . . . . .	56
SPQ.dao.IAccountDAO . . . . .	59
SPQ.dao.AccountDAO . . . . .	23
SPQ.gateway.IEroskiGateway . . . . .	60
SPQ.gateway.Eroski . . . . .	44
SPQ.gateway.IFacebookGateway . . . . .	61
SPQ.gateway.Facebook . . . . .	49
SPQ.gateway.IGoogleGateway . . . . .	62
SPQ.gateway.Google . . . . .	54
SPQ.gateway.IPayPalGateway . . . . .	67
SPQ.gateway.PayPal . . . . .	114
SPQ.dao.IUserDAO . . . . .	68
SPQ.dao.UserDAO . . . . .	158
SPQ.dao.UserDAO . . . . .	158
SPQ.dao.UserDAO . . . . .	158
SPQ.gateway.IVisaGateway . . . . .	69
SPQ.gateway.Visa . . . . .	206
SPQ.main.Main . . . . .	70
SPQ.controller.NeverEmptyController . . . . .	71
SPQ.NeverEmptyServerManager . . . . .	99
SPQ.PaypalServer . . . . .	115
SPQ.dao.ProductDAO . . . . .	125
SPQ.remote.RMIServiceLocator . . . . .	135
SPQ.RMITest . . . . .	137
Thread	

SPQ.EroskiService . . . . .	47
SPQ.FacebookService . . . . .	52
SPQ.GoogleService . . . . .	57
SPQ.PaypalService . . . . .	116
SPQ.VisaService . . . . .	208
SPQ.Utilities . . . . .	180
SPQ.VisaServer . . . . .	207
ActionListener	
SPQ.gui.component.ProductLabel . . . . .	130
SPQ.gui.Ventana MetodoPago . . . . .	184
SPQ.gui.Ventana Pago . . . . .	187
SPQ.gui.Ventana Perfil . . . . .	191
SPQ.gui.Ventana Principal . . . . .	194
SPQ.gui.Ventana Seleccionar Registro . . . . .	201
SPQ.gui.Ventana Ticket . . . . .	203
JFrame	
SPQ.gui.Ventana Inicio . . . . .	181
SPQ.gui.Ventana MetodoPago . . . . .	184
SPQ.gui.Ventana Pago . . . . .	187
SPQ.gui.Ventana Perfil . . . . .	191
SPQ.gui.Ventana Principal . . . . .	194
SPQ.gui.Ventana Registro . . . . .	197
SPQ.gui.Ventana Seleccionar Registro . . . . .	201
SPQ.gui.Ventana Ticket . . . . .	203
JPanel	
SPQ.gui.component.ProductLabel . . . . .	130
Remote	
SPQ.remote.INeverEmptyFacade . . . . .	63
SPQ.NeverEmptyServer . . . . .	90
SPQ.remote.NeverEmptyFacade . . . . .	83
Serializable	
SPQ.data.Product . . . . .	119
SPQ.data.Product . . . . .	119
SPQ.data.User . . . . .	142
SPQ.data.User . . . . .	142
SPQ.data.User . . . . .	142
SPQ.dto.PaymentDTO . . . . .	101
SPQ.dto.PaymentDTO . . . . .	101
SPQ.dto.PaymentDTO . . . . .	101
SPQ.dto.ProductDTO . . . . .	127
SPQ.dto.ProductDTO . . . . .	127
SPQ.dto.UserDTO . . . . .	164
SPQ.dto.UserDTO . . . . .	164
SPQ.dto.UserDTO . . . . .	164
UnicastRemoteObject	
SPQ.NeverEmptyServer . . . . .	90
SPQ.remote.NeverEmptyFacade . . . . .	83

# Chapter 4

## Class Index

### 4.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

SPQ.data.Account . . . . .	19
SPQ.dao.AccountDAO . . . . .	23
SPQ.ClientTest	
ClientTest contains the junit tests of the client side . . . . .	25
SPQ.CoberturaTest . . . . .	32
SPQ.DAOMockTest . . . . .	35
SPQ.DATAProductTest . . . . .	37
SPQ.DATAUserTest . . . . .	39
SPQ.gateway.Eroski . . . . .	44
SPQ.EroskiServer . . . . .	46
SPQ.EroskiService . . . . .	47
SPQ.gateway.Facebook . . . . .	49
SPQ.FacebookServer . . . . .	51
SPQ.FacebookService . . . . .	52
SPQ.gateway.Google . . . . .	54
SPQ.GoogleServer . . . . .	56
SPQ.GoogleService . . . . .	57
SPQ.dao.IAccountDAO . . . . .	59
SPQ.gateway.IEroskiGateway . . . . .	60
SPQ.gateway.IFacebookGateway . . . . .	61
SPQ.gateway.IGoogleGateway . . . . .	62
SPQ.remote.INeverEmptyFacade . . . . .	63
SPQ.gateway.IPayPalGateway . . . . .	67
SPQ.dao.IUserDAO . . . . .	68
SPQ.gateway.IVisaGateway . . . . .	69
SPQ.main.Main	
The Main class contains the main method of the application, the start point of the client execution	70
SPQ.controller.NeverEmptyController	
This class acts as an intermediary between the graphic user interface and the classes that call the server. It mostly contains registration, login and payment methods . . . . .	71
SPQ.remote.NeverEmptyFacade . . . . .	83
SPQ.NeverEmptyServer	
This class contains the methods that register and login a user, and recover information about them . . . . .	90
SPQ.NeverEmptyServerManager	
This class contains the main method to execute the server side . . . . .	99

<a href="#">SPQ.dto.PaymentDTO</a>	This class prepares a payment to be transferred . . . . .	101
<a href="#">SPQ.gateway.PayPal</a>	. . . . .	114
<a href="#">SPQ.PaypalServer</a>	. . . . .	115
<a href="#">SPQ.PaypalService</a>	. . . . .	116
<a href="#">SPQ.data.Product</a>	. . . . .	119
<a href="#">SPQ.dao.ProductDAO</a>	. . . . .	125
<a href="#">SPQ.dto.ProductDTO</a>	This class prepares a product to be transferred . . . . .	127
<a href="#">SPQ.gui.component.ProductLabel</a>	This component extends JPanel. Represents one product. <a href="#">ProductLabel</a> consists of four labels: product name, price, sale (if > 0%), quantity; and two buttons: to increase the quantity by one and to subtract by one . . . . .	130
<a href="#">SPQ.remote.RMIServiceLocator</a>	<a href="#">RMIServiceLocator</a> prepares the connection to the server side by founding the <a href="#">NeverEmptyFacade</a> in the rmi registry . . . . .	135
<a href="#">SPQ.RMITest</a>	. . . . .	137
<a href="#">SPQ.data.User</a>	This class defines a <a href="#">User</a> . . . . .	142
<a href="#">SPQ.dao.UserDAO</a>	. . . . .	158
<a href="#">SPQ.dto.UserDTO</a>	This class prepares a user to be transferred . . . . .	164
<a href="#">SPQ.Utilities</a>	<a href="#">Utilities</a> class. The <a href="#">Utilities</a> class gathers reusable utility methods in different points of the project . . . . .	180
<a href="#">SPQ.gui.VentanaInicio</a>	This is the first window of the application. It contains a form with username and password fields to login. And two buttons: Login and Sign Up. The login button (bLogin) calls the login function of the controller and sends the form data. The sign up (bSignUp) button calls "Ventana← SeleccionarRegistro" window . . . . .	181
<a href="#">SPQ.gui.VentanaMetodoPago</a>	This window contains two main buttons, paypal (bPaypal) button and visa button (bVisa). Both of them call "VentanaPago" but bPayPal passes "paypal" String to "VentanaPago" and bVisa "visa" . . . . .	184
<a href="#">SPQ.gui.VentanaPago</a>	This window shows the last step to make a payment. <a href="#">VentanaPago</a> has two modes: visa and paypal. Depending on the input attribute the form changes. On the top is always shown the shipping address that can be modified in "VentanaPerfil" The PayPal form consist of two fields: username and password. The VISA form consist of four fields: cardnumber, cardholder, expiration date and CVV. The left bottom button allows to use a new paypal user or visa data to pay. It calls "Animation" and transforms the form. The right bottom button calls the NeverEmptyController payWithPayPal(PaymentDTO) or payWithVisa(PaymentDTO) function . . . . .	187
<a href="#">SPQ.gui.VentanaPerfil</a>	<a href="#">VentanaPerfil</a> is a windows that shows the user profile data. It has 4 sections: NeverEmpty profile data, shipping address, VISA data, PayPal data. On the bottom there is a button that allows the form editing . . . . .	191
<a href="#">SPQ.gui.VentanaPrincipal</a>	This is the main window of the application. It contains two lists: the catalog list with all the products and the shoppingcart list . . . . .	194
<a href="#">SPQ.gui.VentanaRegistro</a>	This window shows the form to signUp on NeverEmpty. The form has 4 fields: username, password, repeat password and email. It is the only information needed to sign up in NeverEmpty. When the register button is pressed the NeverEmptyController login function is called, the form data and the register method is passed to the controller . . . . .	197
<a href="#">SPQ.gui.VentanaSeleccionarRegistro</a>	This window shows the options available to sign up. There are three sign up options: Sign up with google, facebook or NeverEmpty . . . . .	201
<a href="#">SPQ.gui.VentanaTicket</a>	This window shows the list of products added to the shopping list, with its respective prices and the total amount in a table . . . . .	203

<a href="#">SPQ.gateway.Visa</a>	.....	206
<a href="#">SPQ.VisaServer</a>	.....	207
<a href="#">SPQ.VisaService</a>	.....	208



# Chapter 5

## File Index

### 5.1 File List

Here is a list of all files with brief descriptions:

EroskiService/src/main/java/SPQ/ <a href="#">EroskiServer.java</a>	214
EroskiService/src/main/java/SPQ/ <a href="#">EroskiService.java</a>	214
EroskiService/src/main/java/SPQ/dao/ <a href="#">ProductDAO.java</a>	213
EroskiService/src/main/java/SPQ/data/ <a href="#">Product.java</a>	213
EroskiService/src/main/java/SPQ/dto/ <a href="#">ProductDTO.java</a>	214
FacebookService/src/main/java/SPQ/ <a href="#">FacebookServer.java</a>	218
FacebookService/src/main/java/SPQ/ <a href="#">FacebookService.java</a>	218
FacebookService/src/main/java/SPQ/dao/ <a href="#">IUserDAO.java</a>	215
FacebookService/src/main/java/SPQ/dao/ <a href="#">UserDAO.java</a>	215
FacebookService/src/main/java/SPQ/data/ <a href="#">User.java</a>	216
FacebookService/src/main/java/SPQ/dto/ <a href="#">UserDTO.java</a>	217
GoogleService/src/main/java/SPQ/ <a href="#">GoogleServer.java</a>	219
GoogleService/src/main/java/SPQ/ <a href="#">GoogleService.java</a>	219
GoogleService/src/main/java/SPQ/dao/ <a href="#">IUserDAO.java</a>	215
GoogleService/src/main/java/SPQ/dao/ <a href="#">UserDAO.java</a>	216
GoogleService/src/main/java/SPQ/data/ <a href="#">User.java</a>	217
GoogleService/src/main/java/SPQ/dto/ <a href="#">UserDTO.java</a>	218
NeverEmptyClient/src/main/java/SPQ/ <a href="#">Utilities.java</a>	223
NeverEmptyClient/src/main/java/SPQ/controller/ <a href="#">NeverEmptyController.java</a>	219
NeverEmptyClient/src/main/java/SPQ/gui/ <a href="#">VentanalInicio.java</a>	220
NeverEmptyClient/src/main/java/SPQ/gui/ <a href="#">VentanaMetodoPago.java</a>	220
NeverEmptyClient/src/main/java/SPQ/gui/ <a href="#">VentanaPago.java</a>	220
NeverEmptyClient/src/main/java/SPQ/gui/ <a href="#">VentanaPerfil.java</a>	221
NeverEmptyClient/src/main/java/SPQ/gui/ <a href="#">VentanaPrincipal.java</a>	221
NeverEmptyClient/src/main/java/SPQ/gui/ <a href="#">VentanaRegistro.java</a>	221
NeverEmptyClient/src/main/java/SPQ/gui/ <a href="#">VentanaSeleccionarRegistro.java</a>	222
NeverEmptyClient/src/main/java/SPQ/gui/ <a href="#">VentanaTicket.java</a>	222
NeverEmptyClient/src/main/java/SPQ/gui/component/ <a href="#">ProductLabel.java</a>	219
NeverEmptyClient/src/main/java/SPQ/main/ <a href="#">Main.java</a>	222
NeverEmptyClient/src/main/java/SPQ/remote/ <a href="#">RMIServiceLocator.java</a>	222
NeverEmptyClient/src/test/java/SPQ/ <a href="#">ClientTest.java</a>	223
NeverEmptyServer/src/main/java/SPQ/ <a href="#">NeverEmptyServer.java</a>	227
NeverEmptyServer/src/main/java/SPQ/ <a href="#">NeverEmptyServerManager.java</a>	227
NeverEmptyServer/src/main/java/SPQ/dao/ <a href="#">UserDAO.java</a>	216
NeverEmptyServer/src/main/java/SPQ/data/ <a href="#">Product.java</a>	213

NeverEmptyServer/src/main/java/SPQ/data/ <a href="#">User.java</a>	217
NeverEmptyServer/src/main/java/SPQ/dto/ <a href="#">PaymentDTO.java</a>	223
NeverEmptyServer/src/main/java/SPQ/dto/ <a href="#">ProductDTO.java</a>	214
NeverEmptyServer/src/main/java/SPQ/dto/ <a href="#">UserDTO.java</a>	218
NeverEmptyServer/src/main/java/SPQ/gateway/ <a href="#">Eroski.java</a>	224
NeverEmptyServer/src/main/java/SPQ/gateway/ <a href="#">Facebook.java</a>	224
NeverEmptyServer/src/main/java/SPQ/gateway/ <a href="#">Google.java</a>	225
NeverEmptyServer/src/main/java/SPQ/gateway/ <a href="#">IEroskiGateway.java</a>	225
NeverEmptyServer/src/main/java/SPQ/gateway/ <a href="#">IFacebookGateway.java</a>	225
NeverEmptyServer/src/main/java/SPQ/gateway/ <a href="#">IGoogleGateway.java</a>	225
NeverEmptyServer/src/main/java/SPQ/gateway/ <a href="#">IPayPalGateway.java</a>	226
NeverEmptyServer/src/main/java/SPQ/gateway/ <a href="#">IVisaGateway.java</a>	226
NeverEmptyServer/src/main/java/SPQ/gateway/ <a href="#">PayPal.java</a>	226
NeverEmptyServer/src/main/java/SPQ/gateway/ <a href="#">Visa.java</a>	226
NeverEmptyServer/src/main/java/SPQ/remote/ <a href="#">INeverEmptyFacade.java</a>	227
NeverEmptyServer/src/main/java/SPQ/remote/ <a href="#">NeverEmptyFacade.java</a>	227
NeverEmptyServer/src/test/java/SPQ/ <a href="#">CoberturaTest.java</a>	228
NeverEmptyServer/src/test/java/SPQ/ <a href="#">DAOMockTest.java</a>	228
NeverEmptyServer/src/test/java/SPQ/ <a href="#">DATAProductTest.java</a>	228
NeverEmptyServer/src/test/java/SPQ/ <a href="#">DATAUserTest.java</a>	228
NeverEmptyServer/src/test/java/SPQ/ <a href="#">RMITest.java</a>	229
PaypalService/src/main/java/SPQ/ <a href="#">PaypalServer.java</a>	229
PaypalService/src/main/java/SPQ/ <a href="#">PaypalService.java</a>	229
PaypalService/src/main/java/SPQ/dao/ <a href="#">IUserDAO.java</a>	215
PaypalService/src/main/java/SPQ/dao/ <a href="#">UserDAO.java</a>	216
PaypalService/src/main/java/SPQ/data/ <a href="#">User.java</a>	217
PaypalService/src/main/java/SPQ/dto/ <a href="#">PaymentDTO.java</a>	224
VisaService/src/main/java/SPQ/ <a href="#">VisaServer.java</a>	230
VisaService/src/main/java/SPQ/ <a href="#">VisaService.java</a>	230
VisaService/src/main/java/SPQ/dao/ <a href="#">AccountDAO.java</a>	229
VisaService/src/main/java/SPQ/dao/ <a href="#">IAccountDAO.java</a>	230
VisaService/src/main/java/SPQ/data/ <a href="#">Account.java</a>	230
VisaService/src/main/java/SPQ/dto/ <a href="#">PaymentDTO.java</a>	224

# Chapter 6

## Namespace Documentation

### 6.1 Package SPQ

Client root package.

#### Packages

- package controller

*This package contains the classes that act as intermediaries between the graphic user interface and the classes that speak with the server.*

- package dao
- package data
- package dto
- package gateway
- package gui

*The SPQ.gui package contains the classes responsible for displaying the graphic user interface of the application.*

- package main

*The SPQ.main contains the main class.*

- package remote

*The SPQ.remote contains the RMI service locator.*

#### Classes

- class ClientTest

*ClientTest contains the junit tests of the client side.*

- class CoberturaTest
- class DAOMockTest
- class DATAProductTest
- class DATAUserTest
- class EroskiServer
- class EroskiService
- class FacebookServer
- class FacebookService
- class GoogleServer
- class GoogleService

- class [NeverEmptyServer](#)  
*This class contains the methods that register and login a user, and recover information about them.*
- class [NeverEmptyServerManager](#)  
*This class contains the main method to execute the server side.*
- class [PaypalServer](#)
- class [PaypalService](#)
- class [RMITest](#)
- class [Utilities](#)  
*Utilities class. The Utilities class gathers reusable utility methods in different points of the project.*
- class [VisaServer](#)
- class [VisaService](#)

### 6.1.1 Detailed Description

Client root package.

This package contains the classes with the main functionality.

This [SPQ](#) package is the client testing root package. Contains the client tests.

## 6.2 Package SPQ.controller

This package contains the classes that act as intermediaries between the graphic user interface and the classes that speak with the server.

### Classes

- class [NeverEmptyController](#)  
*This class acts as an intermediary between the graphic user interface and the classes that call the server. It mostly contains registration, login and payment methods.*

### 6.2.1 Detailed Description

This package contains the classes that act as intermediaries between the graphic user interface and the classes that speak with the server.

## 6.3 Package SPQ.dao

### Classes

- class [AccountDAO](#)
- interface [IAccountDAO](#)
- interface [IUserDAO](#)
- class [ProductDAO](#)
- class [UserDAO](#)

## 6.4 Package SPQ.data

### Classes

- class [Account](#)
- class [Product](#)
- class [User](#)

*This class defines a [User](#).*

### 6.4.1 Detailed Description

This package contains the classes that define the object user and product, their parameters and their constructors

## 6.5 Package SPQ.dto

### Classes

- class [PaymentDTO](#)  
*This class prepares a payment to be transferred.*
- class [ProductDTO](#)  
*This class prepares a product to be transferred.*
- class [UserDTO](#)  
*This class prepares a user to be transferred.*

### 6.5.1 Detailed Description

This package prepare some object to transfer to client

## 6.6 Package SPQ.gateway

### Classes

- class [Eroski](#)
- class [Facebook](#)
- class [Google](#)
- interface [IEroskiGateway](#)
- interface [IFacebookGateway](#)
- interface [IGoogleGateway](#)
- interface [IPayPalGateway](#)
- interface [IVisaGateway](#)
- class [PayPal](#)
- class [Visa](#)

## 6.7 Package SPQ.gui

The [SPQ.gui](#) package contains the classes responsible for displaying the graphic user interface of the application.

### Packages

- package [component](#)

*The [SPQ.gui.component](#) consist of GUI tiny components that are used by other GUI bigger components as windows.*

### Classes

- class [VentanaInicio](#)

*This is the first window of the application. It contains a form with username and password fields to login. And two buttons: Login and Sign Up. The login button (*bLogin*) calls the login function of the controller and sends the form data. The sign up (*bSignUp*) button calls "VentanaSeleccionarRegistro" window.*

- class [Ventana MetodoPago](#)

*This window contains two main buttons, paypal (*bPaypal*) button and visa button (*bVisa*). Both of them call "VentanaPago" but *bPayPal* passes "paypal" String to "VentanaPago" and *bVisa* "visa".*

- class [VentanaPago](#)

*This window shows the last step to make a payment. [VentanaPago](#) has two modes: visa and paypal. Depending on the input attribute the form changes. On the top is always shown the shipping address that can be modified in "VentanaPerfil" The PayPal form consist of two fields: username and password. The VISA form consist of four fields: cardnumber, cardholder, expiration date and CVV. The left bottom button allows to use a new paypal user or visa data to pay. It calls "Animation" and transforms the form. The right bottom button calls the NeverEmptyController *payWithPayPal(PaymentDTO)* or *payWithVisa(PaymentDTO)* function.*

- class [VentanaPerfil](#)

*[VentanaPerfil](#) is a windows that shows the user profile data. It has 4 sections: NeverEmpty profile data, shipping address, VISA data, PayPal data. On the bottom there is a button that allows the form editing.*

- class [VentanaPrincipal](#)

*This is the main window of the application. It contains two lists: the catalog list with all the products and the shoppingcart list.*

- class [VentanaRegistro](#)

*This window shows the form to signUp on NeverEmpty. The form has 4 fields: username, password, repeat password and email. It is the only information needed to sign up in NeverEmpty. When the register button is pressed the NeverEmptyController login function is called, the form data and the register method is passed to the controller.*

- class [VentanaSeleccionarRegistro](#)

*This window shows the options available to sign up. There are three sign up options: Sign up with google, facebook or NeverEmpty.*

- class [VentanaTicket](#)

*This window shows the list of products added to the shopping list, with its respective prices and the total amount in a table.*

### 6.7.1 Detailed Description

The [SPQ.gui](#) package contains the classes responsible for displaying the graphic user interface of the application.

## 6.8 Package SPQ.gui.component

The [SPQ.gui.component](#) consist of GUI tiny components that are used by other GUI bigger components as windows.

## Classes

- class [ProductLabel](#)

*This component extends JPanel. Represents one product. [ProductLabel](#) consists of four labels: product name, price, sale (if > 0%), quantity; and two buttons: to increase the quantity by one and to subtract by one.*

### 6.8.1 Detailed Description

The [SPQ.gui.component](#) consist of GUI tiny components that are used by other GUI bigger components as windows.

## 6.9 Package SPQ.main

The [SPQ.main](#) contains the main class.

## Classes

- class [Main](#)

*The [Main](#) class contains the main method of the application, the start point of the client execution.*

### 6.9.1 Detailed Description

The [SPQ.main](#) contains the main class.

## 6.10 Package SPQ.remote

The [SPQ.remote](#) contains the RMI service locator.

## Classes

- interface [INeverEmptyFacade](#)
- class [NeverEmptyFacade](#)
- class [RMIServiceLocator](#)

*[RMIServiceLocator](#) prepares the connection to the server side by founding the [NeverEmptyFacade](#) in the rmi registry.*

### 6.10.1 Detailed Description

The [SPQ.remote](#) contains the RMI service locator.



# Chapter 7

## Class Documentation

### 7.1 SPQ.data.Account Class Reference

#### Public Member Functions

- `Account` (String cardholder, long cardnumber, int cvv, double balance, String expDate)
- `Account` (`PaymentDTO` paymentDTO)
- String `getCardholder` ()
- void `setCardholder` (String cardholder)
- int `getCvv` ()
- void `setCvv` (int cvv)
- double `getBalance` ()
- void `setBalance` (double balance)
- long `getCardnumber` ()
- void `setCardnumber` (int cardnumber)
- String `getExpDate` ()
- void `setExpDate` (String expDate)

#### 7.1.1 Detailed Description

Definition at line 11 of file Account.java.

#### 7.1.2 Constructor & Destructor Documentation

##### 7.1.2.1 `Account()` [1/2]

```
SPQ.data.Account.Account (
    String cardholder,
    long cardnumber,
    int cvv,
    double balance,
    String expDate )
```

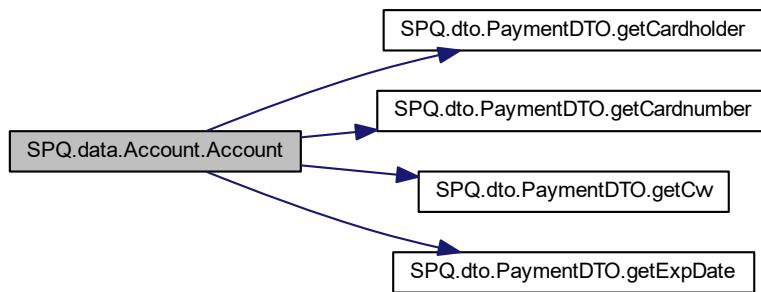
Definition at line 18 of file Account.java.

### 7.1.2.2 Account() [2/2]

```
SPQ.data.Account.Account (
    PaymentDTO paymentDTO )
```

Definition at line 27 of file Account.java.

Here is the call graph for this function:



## 7.1.3 Member Function Documentation

### 7.1.3.1 getBalance()

```
double SPQ.data.Account.getBalance ( )
```

Definition at line 52 of file Account.java.

Here is the caller graph for this function:

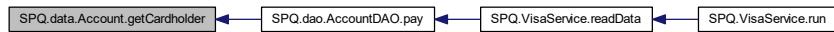


### 7.1.3.2 getCardholder()

```
String SPQ.data.Account.getCardholder ( )
```

Definition at line 36 of file Account.java.

Here is the caller graph for this function:

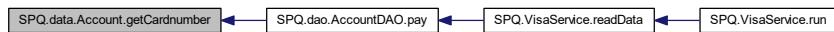


### 7.1.3.3 getCardnumber()

```
long SPQ.data.Account.getCardnumber ( )
```

Definition at line 60 of file Account.java.

Here is the caller graph for this function:



### 7.1.3.4 getCvv()

```
int SPQ.data.Account.getCvv ( )
```

Definition at line 44 of file Account.java.

Here is the caller graph for this function:



### 7.1.3.5 getExpDate()

```
String SPQ.data.Account.getExpDate ( )
```

Definition at line 68 of file Account.java.

Here is the caller graph for this function:



### 7.1.3.6 setBalance()

```
void SPQ.data.Account.setBalance ( double balance )
```

Definition at line 56 of file Account.java.

Here is the caller graph for this function:



### 7.1.3.7 setCardholder()

```
void SPQ.data.Account.setCardholder ( String cardholder )
```

Definition at line 40 of file Account.java.

### 7.1.3.8 setCardnumber()

```
void SPQ.data.Account.setCardnumber ( int cardnumber )
```

Definition at line 64 of file Account.java.

### 7.1.3.9 setCvv()

```
void SPQ.data.Account.setCvv (
    int cvv )
```

Definition at line 48 of file Account.java.

### 7.1.3.10 setExpDate()

```
void SPQ.data.Account.setExpDate (
    String expDate )
```

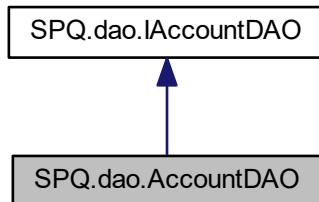
Definition at line 72 of file Account.java.

The documentation for this class was generated from the following file:

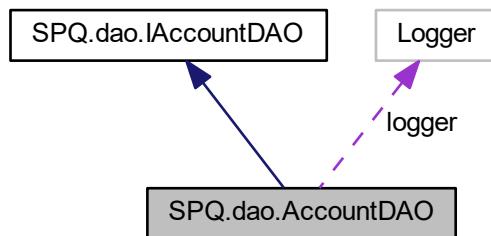
- VisaService/src/main/java/SPQ/data/Account.java

## 7.2 SPQ.dao.AccountDAO Class Reference

Inheritance diagram for SPQ.dao.AccountDAO:



Collaboration diagram for SPQ.dao.AccountDAO:



## Public Member Functions

- `AccountDAO ()`
- boolean `pay (Account account, double total)`
- void `storeAccount (Account account)`

### 7.2.1 Detailed Description

Definition at line 13 of file AccountDAO.java.

### 7.2.2 Constructor & Destructor Documentation

#### 7.2.2.1 AccountDAO()

```
SPQ.dao.AccountDAO.AccountDAO ( )
```

Definition at line 16 of file AccountDAO.java.

### 7.2.3 Member Function Documentation

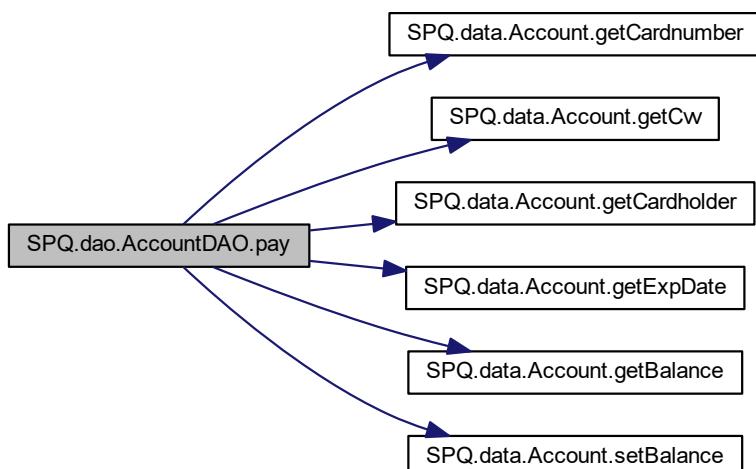
#### 7.2.3.1 pay()

```
boolean SPQ.dao.AccountDAO.pay (
    Account account,
    double total )
```

Implements `SPQ.dao.IAccountDAO`.

Definition at line 20 of file AccountDAO.java.

Here is the call graph for this function:



Here is the caller graph for this function:



#### 7.2.3.2 storeAccount()

```
void SPQ.dao.AccountDAO.storeAccount (
    Account account )
```

Definition at line 65 of file AccountDAO.java.

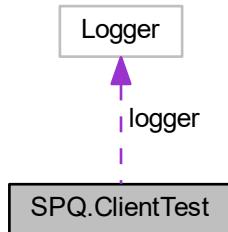
The documentation for this class was generated from the following file:

- VisaService/src/main/java/SPQ/dao/AccountDAO.java

## 7.3 SPQ.ClientTest Class Reference

[ClientTest](#) contains the junit tests of the client side.

Collaboration diagram for SPQ.ClientTest:



## Public Member Functions

- void `setUp()`
- void `testSetUpBadArguments()`
- void `testGetProducts()`
- void `test GetUser()`
- void `testPayWithPayPal()`
- void `testPayWithVisa()`
- void `testUpdateUser()`
- void `restoreUser()`
- void `testRegisterUserNeverEmpty()`
- void `testRegisterUserGoogle()`
- void `testRegisterUserFacebook()`
- void `testLogin()`
- void `testLoginFails()`

### 7.3.1 Detailed Description

[ClientTest](#) contains the junit tests of the client side.

Definition at line 26 of file ClientTest.java.

### 7.3.2 Member Function Documentation

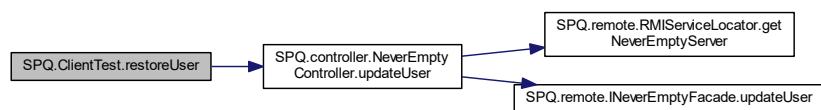
#### 7.3.2.1 `restoreUser()`

```
void SPQ.ClientTest.restoreUser( )
```

Restores a previous user to maintain the database as it was before the testing.

Definition at line 158 of file ClientTest.java.

Here is the call graph for this function:



### 7.3.2.2 `setUp()`

```
void SPQ.ClientTest.setUp ( )
```

Prepares the class for testing instantiating the NeverEmptyController attribute.

Definition at line 36 of file ClientTest.java.

Here is the call graph for this function:



### 7.3.2.3 `testGetProducts()`

```
void SPQ.ClientTest.testGetProducts ( )
```

Tests getting products from the server.

Definition at line 80 of file ClientTest.java.

Here is the call graph for this function:



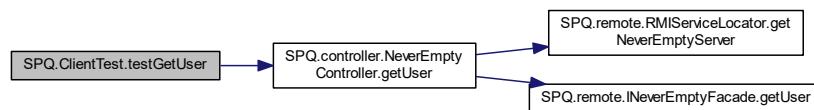
### 7.3.2.4 test GetUser()

```
void SPQ.ClientTest.test GetUser ( )
```

Tests getting a user from the server.

Definition at line 93 of file ClientTest.java.

Here is the call graph for this function:



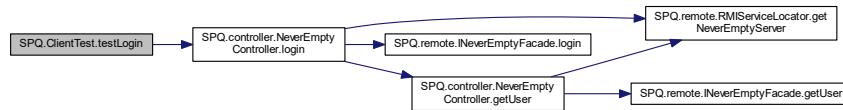
### 7.3.2.5 test Login()

```
void SPQ.ClientTest.test Login ( )
```

Tests a signing in into the application.

Definition at line 213 of file ClientTest.java.

Here is the call graph for this function:



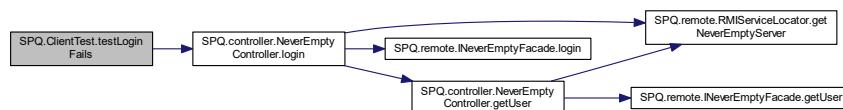
### 7.3.2.6 test LoginFails()

```
void SPQ.ClientTest.test LoginFails ( )
```

Tests a signing in with wrong user data. It is expected to fail.

Definition at line 229 of file ClientTest.java.

Here is the call graph for this function:



### 7.3.2.7 testPayWithPayPal()

```
void SPQ.ClientTest.testPayWithPayPal ( )
```

Tests a payment with PayPal

Definition at line 107 of file ClientTest.java.

Here is the call graph for this function:



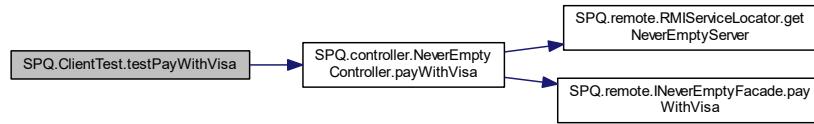
### 7.3.2.8 testPayWithVisa()

```
void SPQ.ClientTest.testPayWithVisa ( )
```

Tests a payment with VISA

Definition at line 121 of file ClientTest.java.

Here is the call graph for this function:



### 7.3.2.9 testRegisterUserFacebook()

```
void SPQ.ClientTest.testRegisterUserFacebook ( )
```

Tests a signing up with Facebook method.

Definition at line 199 of file ClientTest.java.

Here is the call graph for this function:



### 7.3.2.10 testRegisterUserGoogle()

```
void SPQ.ClientTest.testRegisterUserGoogle ( )
```

Tests a signing up with Google method.

Definition at line 185 of file ClientTest.java.

Here is the call graph for this function:



### 7.3.2.11 testRegisterUserNeverEmpty()

```
void SPQ.ClientTest.testRegisterUserNeverEmpty( )
```

Tests a signing up with the NeverEmpty method.

Definition at line 171 of file ClientTest.java.

Here is the call graph for this function:



### 7.3.2.12 testSetUpBadArguments()

```
void SPQ.ClientTest.testSetUpBadArguments( )
```

Tests the exception control of the setting up from RMIServiceLocator passing wrong arguments.

Definition at line 57 of file ClientTest.java.

Here is the call graph for this function:



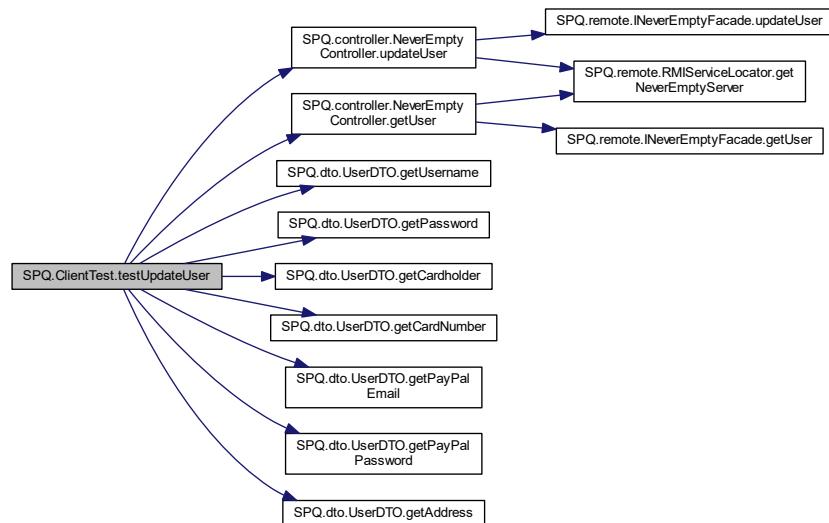
### 7.3.2.13 testUpdateUser()

```
void SPQ.ClientTest.testUpdateUser( )
```

Tests the update of an existing user.

Definition at line 135 of file ClientTest.java.

Here is the call graph for this function:

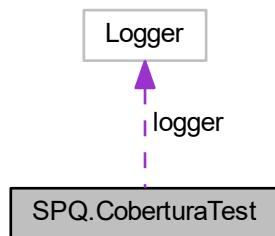


The documentation for this class was generated from the following file:

- NeverEmptyClient/src/test/java/SPQ/[ClientTest.java](#)

## 7.4 SPQ.CoberturaTest Class Reference

Collaboration diagram for SPQ.CoberturaTest:



## Public Member Functions

- void [test \(\)](#)

### 7.4.1 Detailed Description

Definition at line 18 of file CoberturaTest.java.

### 7.4.2 Member Function Documentation

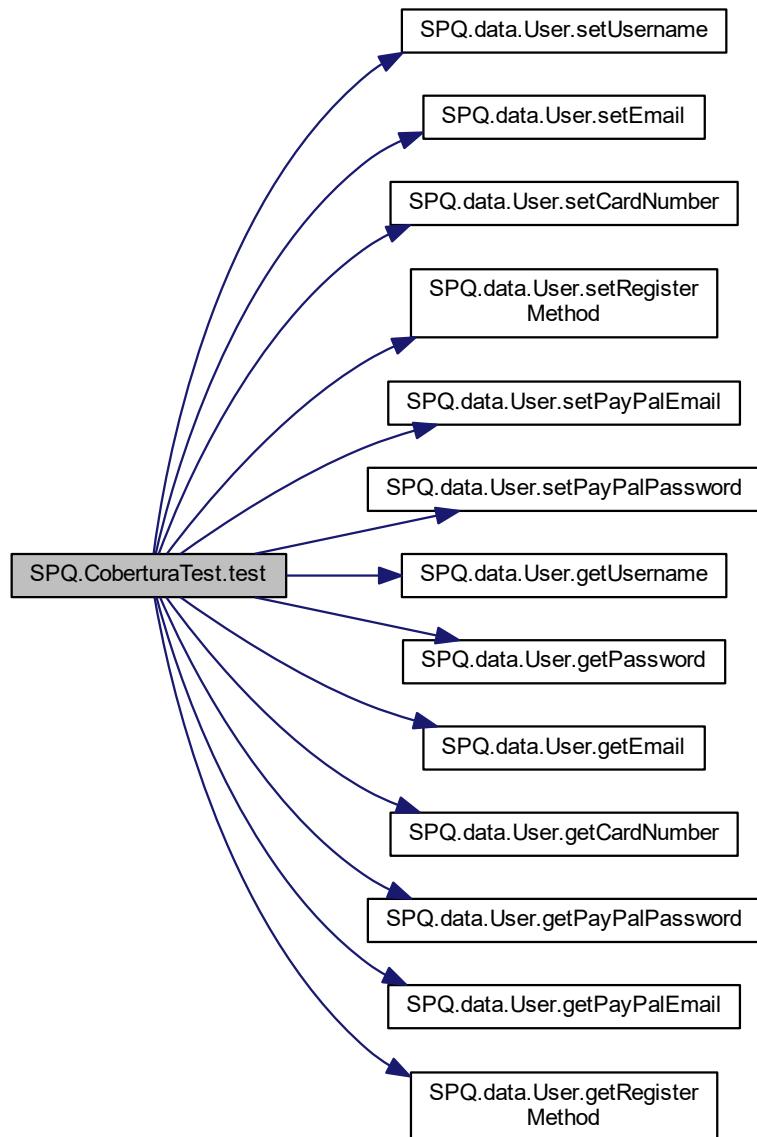
#### 7.4.2.1 [test\(\)](#)

```
void SPQ.CoberturaTest.test ( )
```

Definition at line 22 of file CoberturaTest.java.

---

Here is the call graph for this function:

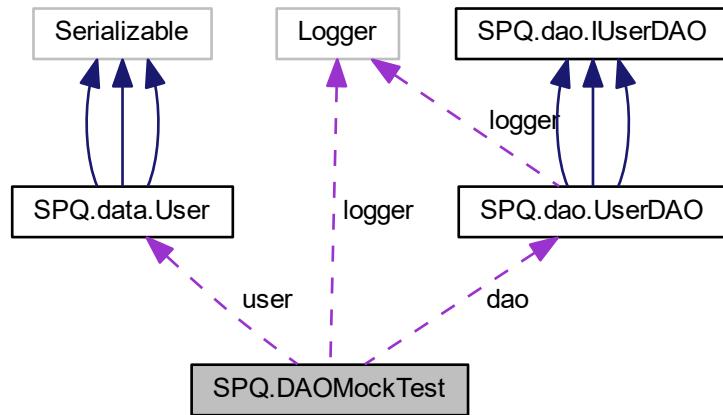


The documentation for this class was generated from the following file:

- NeverEmptyServer/src/test/java/SPQ/CoberturaTest.java

## 7.5 SPQ.DAOMockTest Class Reference

Collaboration diagram for SPQ.DAOMockTest:



### Public Member Functions

- void [setUp \(\) throws Exception](#)
- void [testRegisterUserCorrectly \(\)](#)
- void [testRegisterUserAlreadyExists \(\)](#)

### Static Public Member Functions

- static junit.framework.Test [suite \(\)](#)

#### 7.5.1 Detailed Description

Definition at line 22 of file DAOMockTest.java.

#### 7.5.2 Member Function Documentation

##### 7.5.2.1 [setUp\(\)](#)

```
void SPQ.DAOMockTest.setUp ( ) throws Exception
```

Definition at line 36 of file DAOMockTest.java.

### 7.5.2.2 suite()

```
static junit.framework.Test SPQ.DAOMockTest.suite ( ) [static]
```

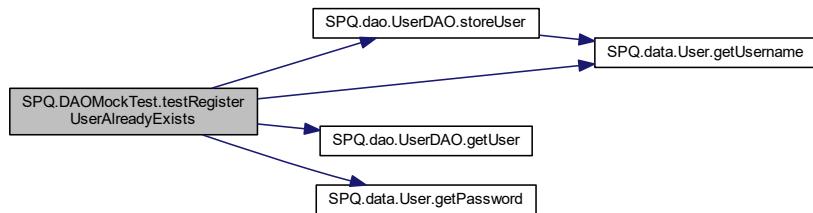
Definition at line 31 of file DAOMockTest.java.

### 7.5.2.3 testRegisterUserAlreadyExists()

```
void SPQ.DAOMockTest.testRegisterUserAlreadyExists ( )
```

Definition at line 61 of file DAOMockTest.java.

Here is the call graph for this function:

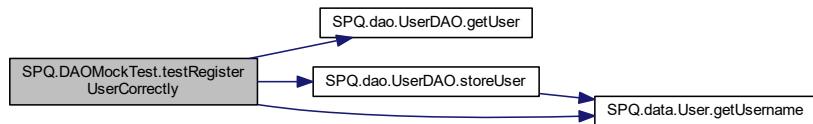


### 7.5.2.4 testRegisterUserCorrectly()

```
void SPQ.DAOMockTest.testRegisterUserCorrectly ( )
```

Definition at line 42 of file DAOMockTest.java.

Here is the call graph for this function:

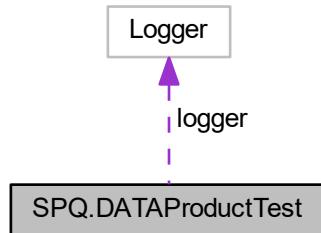


The documentation for this class was generated from the following file:

- NeverEmptyServer/src/test/java/SPQ/DAOMockTest.java

## 7.6 SPQ.DATAProductTest Class Reference

Collaboration diagram for SPQ.DATAProductTest:



### Public Member Functions

- void `setUp () throws Exception`
- void `ProductTest ()`
- void `ProductDTOTest ()`

#### 7.6.1 Detailed Description

Definition at line 18 of file `DATAProductTest.java`.

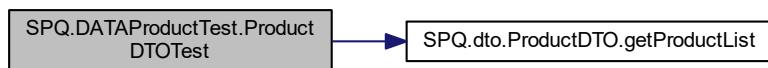
#### 7.6.2 Member Function Documentation

##### 7.6.2.1 ProductDTOTest()

```
void SPQ.DATAProductTest.ProductDTOTest ( )
```

Definition at line 47 of file `DATAProductTest.java`.

Here is the call graph for this function:

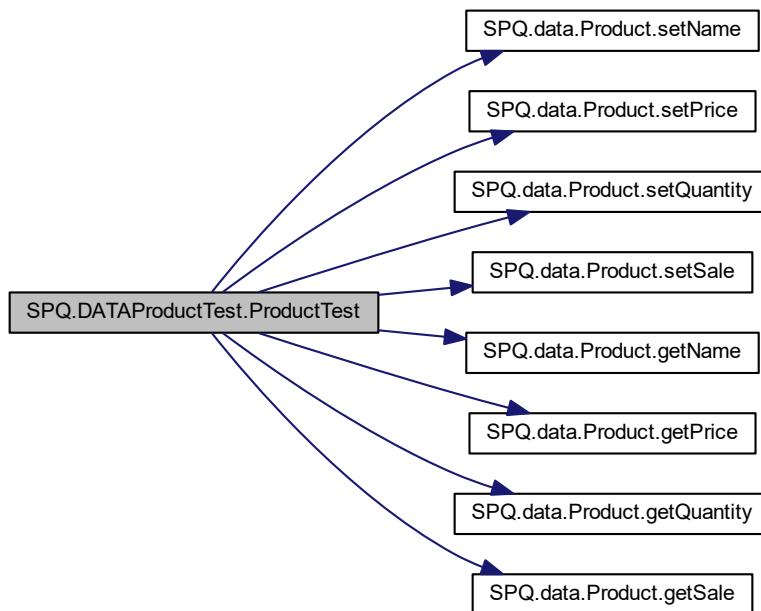


### 7.6.2.2 ProductTest()

```
void SPQ.DATAProductTest.ProductTest( )
```

Definition at line 30 of file DATAProductTest.java.

Here is the call graph for this function:



### 7.6.2.3 setUp()

```
void SPQ.DATAProductTest.setUp( ) throws Exception
```

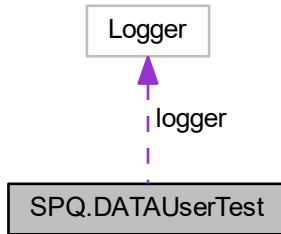
Definition at line 24 of file DATAProductTest.java.

The documentation for this class was generated from the following file:

- [NeverEmptyServer/src/test/java/SPQ/DATAProductTest.java](#)

## 7.7 SPQ.DATAUserTest Class Reference

Collaboration diagram for SPQ.DATAUserTest:



### Public Member Functions

- void [setUp \(\) throws Exception](#)
- void [Jesusitest \(\)](#)
- void [Alvarotest \(\)](#)
- void [Enaratest \(\)](#)
- void [Cristiantest \(\)](#)
- void [Ekaitztest \(\)](#)
- void [UserDTOTest \(\)](#)

#### 7.7.1 Detailed Description

Definition at line 11 of file `DATAUserTest.java`.

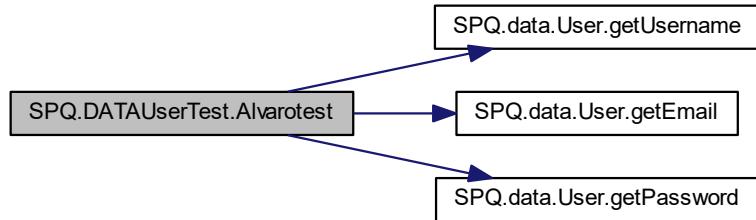
#### 7.7.2 Member Function Documentation

##### 7.7.2.1 Alvarotest()

```
void SPQ.DATAUserTest.Alvarotest ( )
```

Definition at line 44 of file `DATAUserTest.java`.

Here is the call graph for this function:

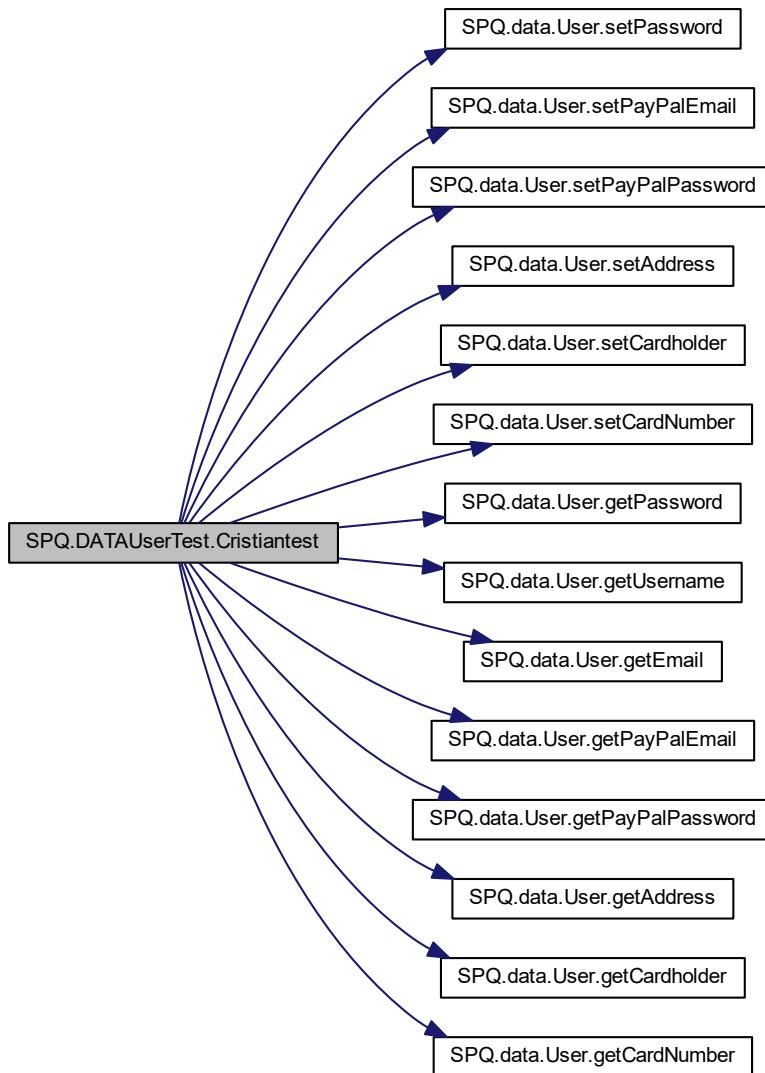


#### 7.7.2.2 Cristiantest()

```
void SPQ.DATAUserTest.Cristiantest ( )
```

Definition at line 60 of file DATAUserTest.java.

Here is the call graph for this function:

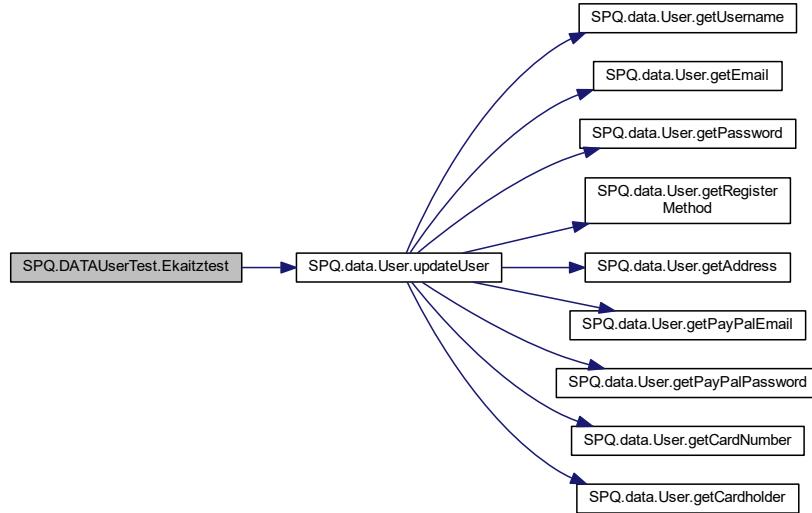


### 7.7.2.3 Ekaitztest()

```
void SPQ.DATAUserTest.Ekaitztest ( )
```

Definition at line 84 of file DATAUserTest.java.

Here is the call graph for this function:

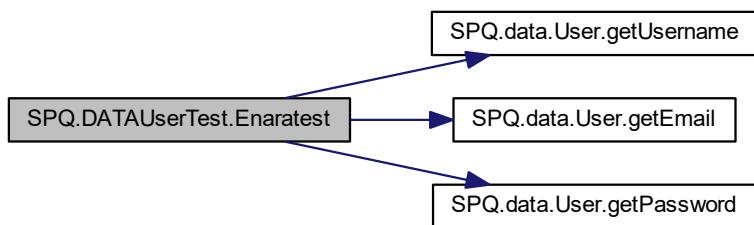


#### 7.7.2.4 Enaratest()

```
void SPQ.DATAUserTest.Enaratest ( )
```

Definition at line 52 of file DATAUserTest.java.

Here is the call graph for this function:

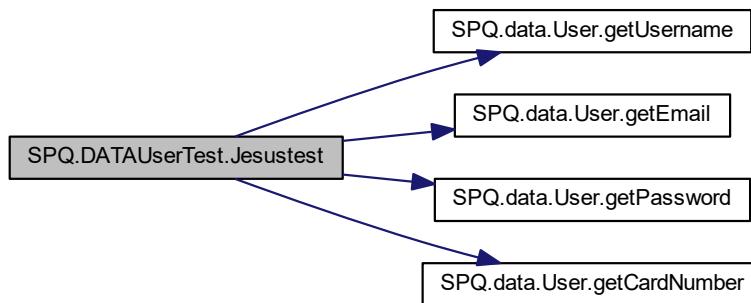


### 7.7.2.5 Jesustest()

```
void SPQ.DATAUserTest.Jesustest ( )
```

Definition at line 34 of file DATAUserTest.java.

Here is the call graph for this function:

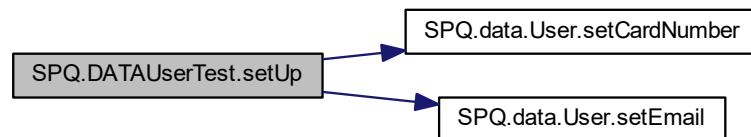


### 7.7.2.6 setUp()

```
void SPQ.DATAUserTest.setUp ( ) throws Exception
```

Definition at line 22 of file DATAUserTest.java.

Here is the call graph for this function:



### 7.7.2.7 UserDTOTest()

```
void SPQ.DATAUserTest.UserDTOTest( )
```

Definition at line 94 of file DATAUserTest.java.

Here is the call graph for this function:

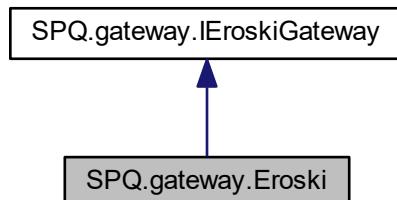


The documentation for this class was generated from the following file:

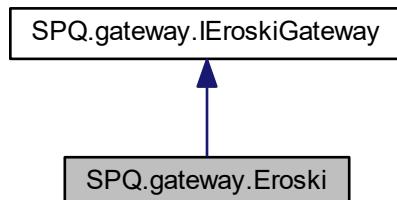
- [NeverEmptyServer/src/test/java/SPQ/DATAUserTest.java](#)

## 7.8 SPQ.gateway.Eroski Class Reference

Inheritance diagram for SPQ.gateway.Eroski:



Collaboration diagram for SPQ.gateway.Eroski:



## Public Member Functions

- [Eroski \(String serverIP, String serverPort\)](#)
- [ProductDTO getProducts \(\)](#)

### 7.8.1 Detailed Description

Definition at line 12 of file Eroski.java.

### 7.8.2 Constructor & Destructor Documentation

#### 7.8.2.1 Eroski()

```
SPQ.gateway.Eroski.Eroski (
    String serverIP,
    String serverPort )
```

Definition at line 18 of file Eroski.java.

### 7.8.3 Member Function Documentation

#### 7.8.3.1 getProducts()

```
ProductDTO SPQ.gateway.Eroski.getProducts ( )
```

Implements [SPQ.gateway.IEroskiGateway](#).

Definition at line 24 of file Eroski.java.

Here is the call graph for this function:



Here is the caller graph for this function:

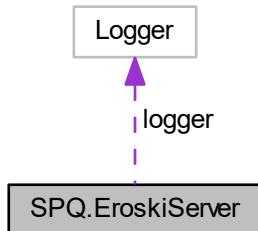


The documentation for this class was generated from the following file:

- [NeverEmptyServer/src/main/java/SPQ/gateway/Eroski.java](#)

## 7.9 SPQ.EroskiServer Class Reference

Collaboration diagram for SPQ.EroskiServer:



### Static Public Member Functions

- static void [main](#) (String args[ ])

#### 7.9.1 Detailed Description

Definition at line 8 of file EroskiServer.java.

#### 7.9.2 Member Function Documentation

##### 7.9.2.1 [main\(\)](#)

```
static void SPQ.EroskiServer.main (
    String args[] ) [static]
```

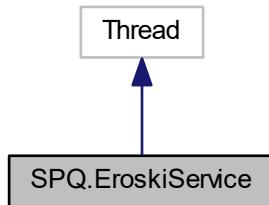
Definition at line 14 of file EroskiServer.java.

The documentation for this class was generated from the following file:

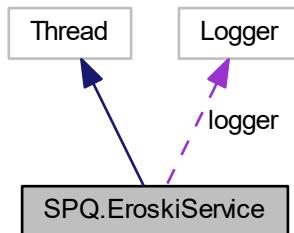
- EroskiService/src/main/java/SPQ/[EroskiServer.java](#)

## 7.10 SPQ.EroskiService Class Reference

Inheritance diagram for SPQ.EroskiService:



Collaboration diagram for SPQ.EroskiService:



### Public Member Functions

- [EroskiService](#) (Socket socket)
- void [run \(\)](#)
- [ProductDTO readData](#) (String data)

#### 7.10.1 Detailed Description

Definition at line 16 of file EroskiService.java.

#### 7.10.2 Constructor & Destructor Documentation

#### 7.10.2.1 EroskiService()

```
SPQ.EroskiService.EroskiService (
    Socket socket )
```

Definition at line 22 of file EroskiService.java.

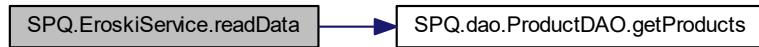
### 7.10.3 Member Function Documentation

#### 7.10.3.1 readData()

```
ProductDTO SPQ.EroskiService.readData (
    String data )
```

Definition at line 56 of file EroskiService.java.

Here is the call graph for this function:



Here is the caller graph for this function:

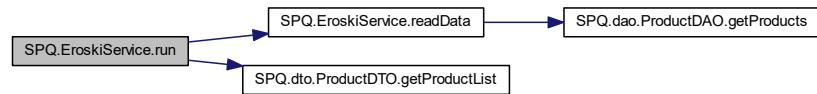


### 7.10.3.2 run()

```
void SPQ.EroskiService.run ( )
```

Definition at line 33 of file EroskiService.java.

Here is the call graph for this function:

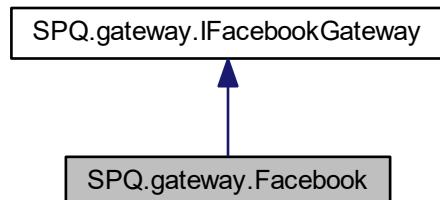


The documentation for this class was generated from the following file:

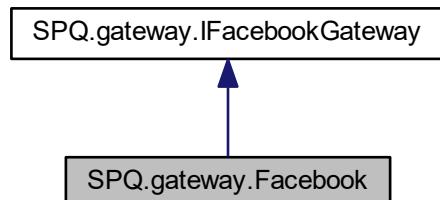
- EroskiService/src/main/java/SPQ/[EroskiService.java](#)

## 7.11 SPQ.gateway.Facebook Class Reference

Inheritance diagram for SPQ.gateway.Facebook:



Collaboration diagram for SPQ.gateway.Facebook:



## Public Member Functions

- [Facebook](#) (String serverIP, String serverPort)
- String [register](#) ([UserDTO](#) userDTO)

### 7.11.1 Detailed Description

Definition at line 14 of file Facebook.java.

### 7.11.2 Constructor & Destructor Documentation

#### 7.11.2.1 Facebook()

```
SPQ.gateway.Facebook.Facebook (
    String serverIP,
    String serverPort )
```

Definition at line 20 of file Facebook.java.

### 7.11.3 Member Function Documentation

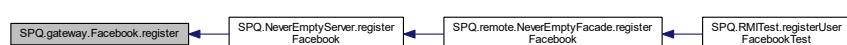
#### 7.11.3.1 register()

```
String SPQ.gateway.Facebook.register (
    UserDTO userDTO )
```

Implements [SPQ.gateway.IFacebookGateway](#).

Definition at line 31 of file Facebook.java.

Here is the caller graph for this function:

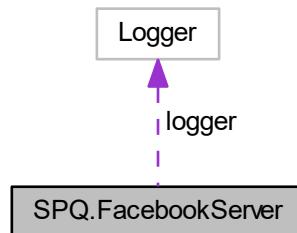


The documentation for this class was generated from the following file:

- [NeverEmptyServer/src/main/java/SPQ/gateway/Facebook.java](#)

## 7.12 SPQ.FacebookServer Class Reference

Collaboration diagram for SPQ.FacebookServer:



### Static Public Member Functions

- static void [main](#) (String args[])

#### 7.12.1 Detailed Description

Definition at line 8 of file FacebookServer.java.

#### 7.12.2 Member Function Documentation

##### 7.12.2.1 main()

```
static void SPQ.FacebookServer.main (
    String args[] ) [static]
```

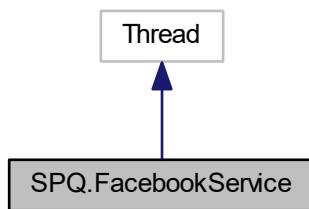
Definition at line 12 of file FacebookServer.java.

The documentation for this class was generated from the following file:

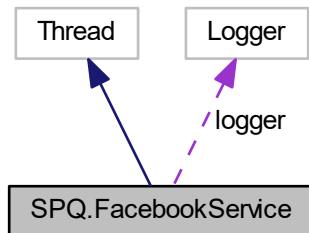
- FacebookService/src/main/java/SPQ/[FacebookServer.java](#)

## 7.13 SPQ.FacebookService Class Reference

Inheritance diagram for SPQ.FacebookService:



Collaboration diagram for SPQ.FacebookService:



### Public Member Functions

- [FacebookService \(Socket socket\)](#)
- [void run \(\)](#)
- [String readData \(UserDTO userDTO\)](#)

#### 7.13.1 Detailed Description

Definition at line 15 of file FacebookService.java.

#### 7.13.2 Constructor & Destructor Documentation

### 7.13.2.1 FacebookService()

```
SPQ.FacebookService.FacebookService (
    Socket socket )
```

Definition at line 21 of file FacebookService.java.

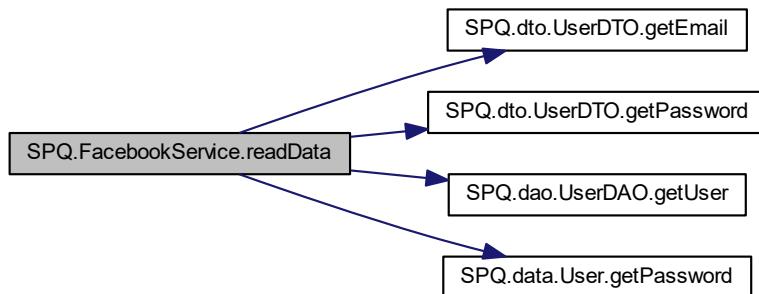
## 7.13.3 Member Function Documentation

### 7.13.3.1 readData()

```
String SPQ.FacebookService.readData (
    UserDTO userDTO )
```

Definition at line 54 of file FacebookService.java.

Here is the call graph for this function:



Here is the caller graph for this function:

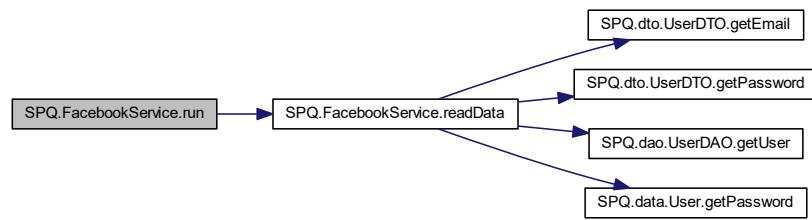


### 7.13.3.2 run()

```
void SPQ.FacebookService.run ( )
```

Definition at line 32 of file FacebookService.java.

Here is the call graph for this function:

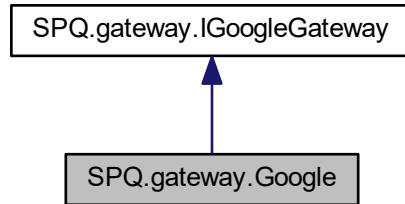


The documentation for this class was generated from the following file:

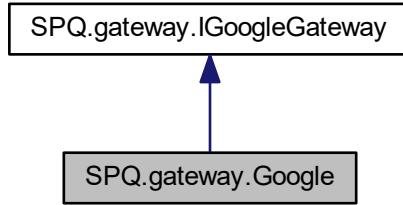
- `FacebookService/src/main/java/SPQ/`[FacebookService.java](#)

## 7.14 SPQ.gateway.Google Class Reference

Inheritance diagram for SPQ.gateway.Google:



Collaboration diagram for SPQ.gateway.Google:



## Public Member Functions

- [Google](#) (String serverIP, String serverPort)
- String [register](#) ([UserDTO](#) userDTO)

### 7.14.1 Detailed Description

Definition at line 13 of file Google.java.

### 7.14.2 Constructor & Destructor Documentation

#### 7.14.2.1 Google()

```
SPQ.gateway.Google.Google (
    String serverIP,
    String serverPort )
```

Definition at line 18 of file Google.java.

### 7.14.3 Member Function Documentation

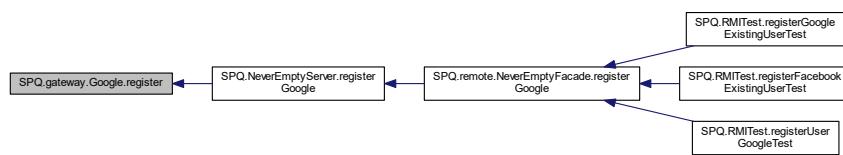
#### 7.14.3.1 register()

```
String SPQ.gateway.Google.register (
    UserDTO userDTO )
```

Implements [SPQ.gateway.IGoogleGateway](#).

Definition at line 29 of file [Google.java](#).

Here is the caller graph for this function:

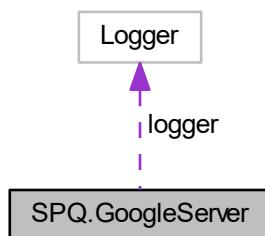


The documentation for this class was generated from the following file:

- [NeverEmptyServer/src/main/java/SPQ/gateway/Google.java](#)

## 7.15 SPQ.GoogleServer Class Reference

Collaboration diagram for SPQ.GoogleServer:



#### Static Public Member Functions

- static void [main](#) (String args[])

#### 7.15.1 Detailed Description

Definition at line 8 of file [GoogleServer.java](#).

### 7.15.2 Member Function Documentation

#### 7.15.2.1 main()

```
static void SPQ.GoogleServer.main (
    String args[] ) [static]
```

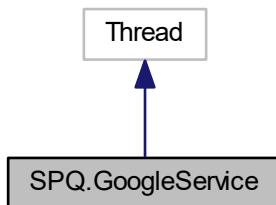
Definition at line 12 of file GoogleServer.java.

The documentation for this class was generated from the following file:

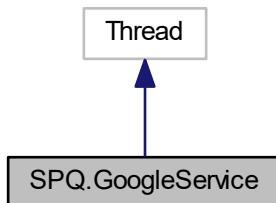
- GoogleService/src/main/java/SPQ/[GoogleServer.java](#)

## 7.16 SPQ.GoogleService Class Reference

Inheritance diagram for SPQ.GoogleService:



Collaboration diagram for SPQ.GoogleService:



## Public Member Functions

- `GoogleService (Socket socket)`
- `void run ()`
- `String readData (UserDTO userDTO)`

### 7.16.1 Detailed Description

Definition at line 13 of file `GoogleService.java`.

### 7.16.2 Constructor & Destructor Documentation

#### 7.16.2.1 `GoogleService()`

```
SPQ.GoogleService.GoogleService (
    Socket socket )
```

Definition at line 18 of file `GoogleService.java`.

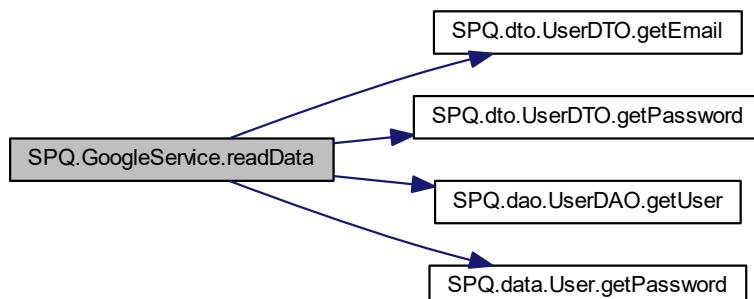
### 7.16.3 Member Function Documentation

#### 7.16.3.1 `readData()`

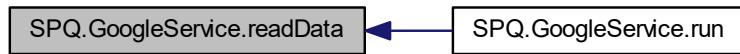
```
String SPQ.GoogleService.readData (
    UserDTO userDTO )
```

Definition at line 51 of file `GoogleService.java`.

Here is the call graph for this function:



Here is the caller graph for this function:

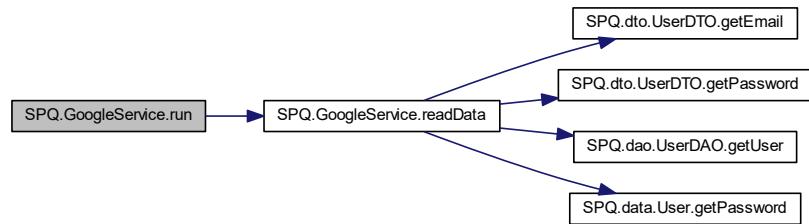


#### 7.16.3.2 run()

```
void SPQ.GoogleService.run ( )
```

Definition at line 29 of file [GoogleService.java](#).

Here is the call graph for this function:

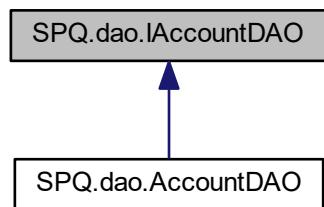


The documentation for this class was generated from the following file:

- [GoogleService/src/main/java/SPQ/GoogleService.java](#)

## 7.17 SPQ.dao.IAccountDAO Interface Reference

Inheritance diagram for SPQ.dao.IAccountDAO:



## Public Member Functions

- boolean [pay \(Account account, double total\)](#)

### 7.17.1 Detailed Description

Definition at line 5 of file [IAccountDAO.java](#).

### 7.17.2 Member Function Documentation

#### 7.17.2.1 pay()

```
boolean SPQ.dao.IAccountDAO.pay (
    Account account,
    double total )
```

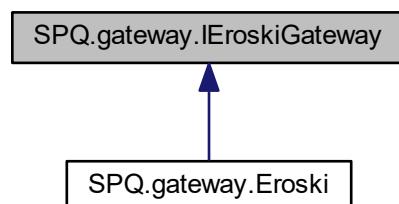
Implemented in [SPQ.dao.AccountDAO](#).

The documentation for this interface was generated from the following file:

- [VisaService/src/main/java/SPQ/dao/IAccountDAO.java](#)

## 7.18 SPQ.gateway.IEroskiGateway Interface Reference

Inheritance diagram for SPQ.gateway.IEroskiGateway:



## Public Member Functions

- [ProductDTO getProducts \(\)](#)

### 7.18.1 Detailed Description

Definition at line 5 of file IEroskiGateway.java.

### 7.18.2 Member Function Documentation

#### 7.18.2.1 getProducts()

```
ProductDTO SPQ.gateway.IEroskiGateway.getProducts( )
```

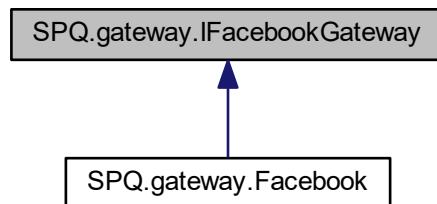
Implemented in [SPQ.gateway.Eroski](#).

The documentation for this interface was generated from the following file:

- [NeverEmptyServer/src/main/java/SPQ/gateway/IEroskiGateway.java](#)

## 7.19 SPQ.gateway.IFacebookGateway Interface Reference

Inheritance diagram for SPQ.gateway.IFacebookGateway:



### Public Member Functions

- String [register \(UserDTO userDTO\)](#)

### 7.19.1 Detailed Description

Definition at line 5 of file IFacebookGateway.java.

### 7.19.2 Member Function Documentation

#### 7.19.2.1 register()

```
String SPQ.gateway.IFacebookGateway.register (
    UserDTO userDTO )
```

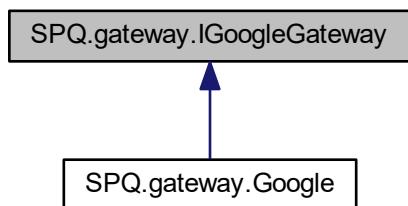
Implemented in [SPQ.gateway.Facebook](#).

The documentation for this interface was generated from the following file:

- [NeverEmptyServer/src/main/java/SPQ/gateway/IFacebookGateway.java](#)

## 7.20 SPQ.gateway.IGoogleGateway Interface Reference

Inheritance diagram for SPQ.gateway.IGoogleGateway:



### Public Member Functions

- String [register \(UserDTO userDTO\)](#)

#### 7.20.1 Detailed Description

Definition at line 5 of file [IGoogleGateway.java](#).

#### 7.20.2 Member Function Documentation

### 7.20.2.1 register()

```
String SPQ.gateway.IGoogleGateway.register (
    UserDTO userDTO )
```

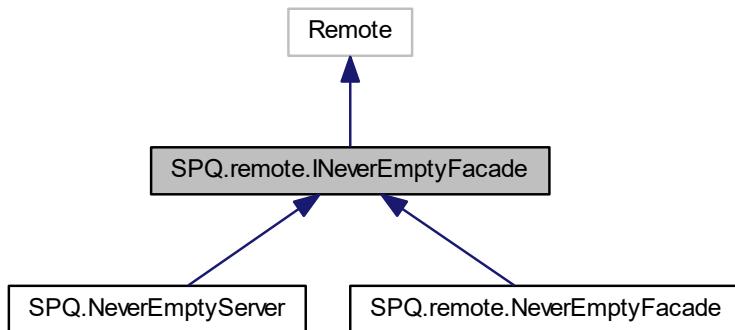
Implemented in [SPQ.gateway.Google](#).

The documentation for this interface was generated from the following file:

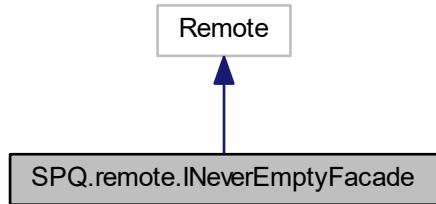
- NeverEmptyServer/src/main/java/SPQ/gateway/[IGoogleGateway.java](#)

## 7.21 SPQ.remote.INeverEmptyFacade Interface Reference

Inheritance diagram for SPQ.remote.INeverEmptyFacade:



Collaboration diagram for SPQ.remote.INeverEmptyFacade:



## Public Member Functions

- boolean `registerGoogle (UserDTO userDTO) throws RemoteException`
- boolean `registerFacebook (UserDTO userDTO) throws RemoteException`
- boolean `registerNeverEmpty (UserDTO userDTO) throws RemoteException`
- boolean `login (UserDTO user) throws RemoteException`
- `ProductDTO getProducts () throws RemoteException`
- `UserDTO getUser (UserDTO user) throws RemoteException`
- boolean `updateUser (UserDTO userDTO) throws RemoteException`
- boolean `payWithPaypal (PaymentDTO paymentDTO) throws RemoteException`
- boolean `payWithVisa (PaymentDTO paymentDTO) throws RemoteException`

### 7.21.1 Detailed Description

Definition at line 11 of file `INeverEmptyFacade.java`.

### 7.21.2 Member Function Documentation

#### 7.21.2.1 `getProducts()`

`ProductDTO SPQ.remote.INeverEmptyFacade.getProducts ( ) throws RemoteException`

Implemented in [SPQ.NeverEmptyServer](#), and [SPQ.remote.NeverEmptyFacade](#).

Here is the caller graph for this function:

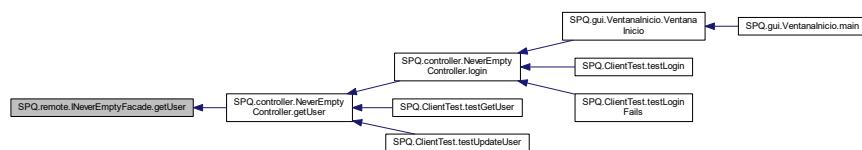


#### 7.21.2.2 `getUser()`

`UserDTO SPQ.remote.INeverEmptyFacade.getUser ( UserDTO user ) throws RemoteException`

Implemented in [SPQ.NeverEmptyServer](#), and [SPQ.remote.NeverEmptyFacade](#).

Here is the caller graph for this function:

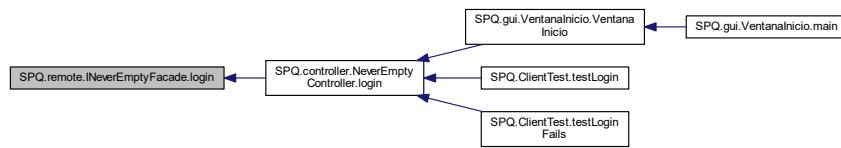


### 7.21.2.3 login()

```
boolean SPQ.remote.INeverEmptyFacade.login (
    UserDTO user ) throws RemoteException
```

Implemented in [SPQ.NeverEmptyServer](#), and [SPQ.remote.NeverEmptyFacade](#).

Here is the caller graph for this function:

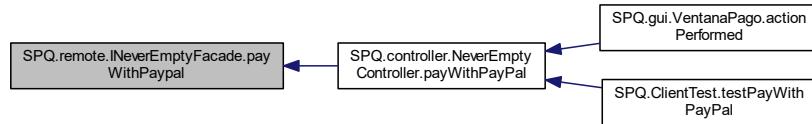


### 7.21.2.4 payWithPaypal()

```
boolean SPQ.remote.INeverEmptyFacade.payWithPaypal (
    PaymentDTO paymentDTO ) throws RemoteException
```

Implemented in [SPQ.NeverEmptyServer](#), and [SPQ.remote.NeverEmptyFacade](#).

Here is the caller graph for this function:

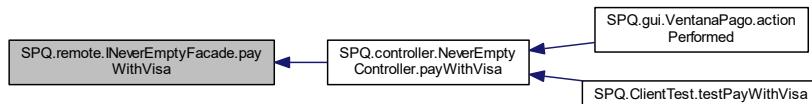


### 7.21.2.5 payWithVisa()

```
boolean SPQ.remote.INeverEmptyFacade.payWithVisa (
    PaymentDTO paymentDTO ) throws RemoteException
```

Implemented in [SPQ.NeverEmptyServer](#), and [SPQ.remote.NeverEmptyFacade](#).

Here is the caller graph for this function:



### 7.21.2.6 registerFacebook()

```
boolean SPQ.remote.INeverEmptyFacade.registerFacebook (
    UserDTO userDTO ) throws RemoteException
```

Implemented in [SPQ.NeverEmptyServer](#), and [SPQ.remote.NeverEmptyFacade](#).

Here is the caller graph for this function:



### 7.21.2.7 registerGoogle()

```
boolean SPQ.remote.INeverEmptyFacade.registerGoogle (
    UserDTO userDTO ) throws RemoteException
```

Implemented in [SPQ.NeverEmptyServer](#), and [SPQ.remote.NeverEmptyFacade](#).

Here is the caller graph for this function:



### 7.21.2.8 registerNeverEmpty()

```
boolean SPQ.remote.INeverEmptyFacade.registerNeverEmpty (
    UserDTO userDTO ) throws RemoteException
```

Implemented in [SPQ.NeverEmptyServer](#), and [SPQ.remote.NeverEmptyFacade](#).

Here is the caller graph for this function:

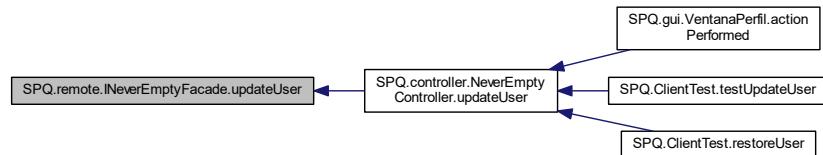


### 7.21.2.9 updateUser()

```
boolean SPQ.remote.INeverEmptyFacade.updateUser (
    UserDTO userDTO ) throws RemoteException
```

Implemented in [SPQ.NeverEmptyServer](#), and [SPQ.remote.NeverEmptyFacade](#).

Here is the caller graph for this function:

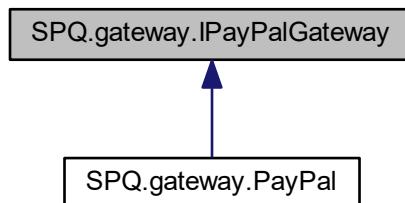


The documentation for this interface was generated from the following file:

- NeverEmptyServer/src/main/java/SPQ/remote/[INeverEmptyFacade.java](#)

## 7.22 SPQ.gateway.IPayPalGateway Interface Reference

Inheritance diagram for SPQ.gateway.IPayPalGateway:



### Public Member Functions

- boolean [pay \(PaymentDTO paymentDTO\)](#)

#### 7.22.1 Detailed Description

Definition at line 5 of file `IPayPalGateway.java`.

## 7.22.2 Member Function Documentation

### 7.22.2.1 pay()

```
boolean SPQ.gateway.IPayPalGateway.pay (
    PaymentDTO paymentDTO )
```

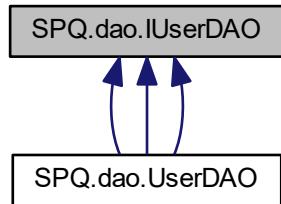
Implemented in [SPQ.gateway.PayPal](#).

The documentation for this interface was generated from the following file:

- [NeverEmptyServer/src/main/java/SPQ/gateway/IPayPalGateway.java](#)

## 7.23 SPQ.dao.IUserDAO Interface Reference

Inheritance diagram for SPQ.dao.IUserDAO:



### Public Member Functions

- [User getUser \(String email\)](#)
- [User getUser \(String email\)](#)
- boolean [pay \(User user, double total\)](#)
- void [storeUser \(User user\)](#)

### 7.23.1 Detailed Description

Definition at line 5 of file [IUserDAO.java](#).

## 7.23.2 Member Function Documentation

### 7.23.2.1 getUser() [1/2]

```
User SPQ.dao.IUserDAO.getUser (
    String email )
```

Implemented in [SPQ.dao.UserDAO](#), and [SPQ.dao.UserDAO](#).

### 7.23.2.2 getUser() [2/2]

```
User SPQ.dao.IUserDAO.getUser (
    String email )
```

Implemented in [SPQ.dao.UserDAO](#), and [SPQ.dao.UserDAO](#).

### 7.23.2.3 pay()

```
boolean SPQ.dao.IUserDAO.pay (
    User user,
    double total )
```

Implemented in [SPQ.dao.UserDAO](#).

### 7.23.2.4 storeUser()

```
void SPQ.dao.IUserDAO.storeUser (
    User user )
```

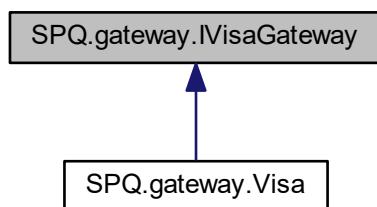
Implemented in [SPQ.dao.UserDAO](#), and [SPQ.dao.UserDAO](#).

The documentation for this interface was generated from the following file:

- [FacebookService/src/main/java/SPQ/dao/IUserDAO.java](#)

## 7.24 SPQ.gateway.IVisaGateway Interface Reference

Inheritance diagram for SPQ.gateway.IVisaGateway:



## Public Member Functions

- boolean [pay \(PaymentDTO paymentDTO\)](#)

### 7.24.1 Detailed Description

Definition at line 5 of file [IVisaGateway.java](#).

### 7.24.2 Member Function Documentation

#### 7.24.2.1 pay()

```
boolean SPQ.gateway.IVisaGateway.pay (
    PaymentDTO paymentDTO )
```

Implemented in [SPQ.gateway.Visa](#).

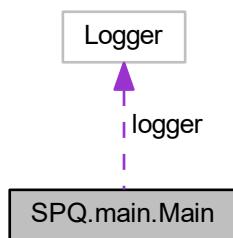
The documentation for this interface was generated from the following file:

- [NeverEmptyServer/src/main/java/SPQ/gateway/IVisaGateway.java](#)

## 7.25 SPQ.main.Main Class Reference

The [Main](#) class contains the main method of the application, the start point of the client execution.

Collaboration diagram for SPQ.main.Main:



## Static Public Member Functions

- static void [main \(String\[ \] args\)](#)

### 7.25.1 Detailed Description

The [Main](#) class contains the main method of the application, the start point of the client execution.

Definition at line 18 of file Main.java.

### 7.25.2 Member Function Documentation

#### 7.25.2.1 main()

```
static void SPQ.main.Main.main (
    String [] args ) [static]
```

Definition at line 22 of file Main.java.

Here is the call graph for this function:



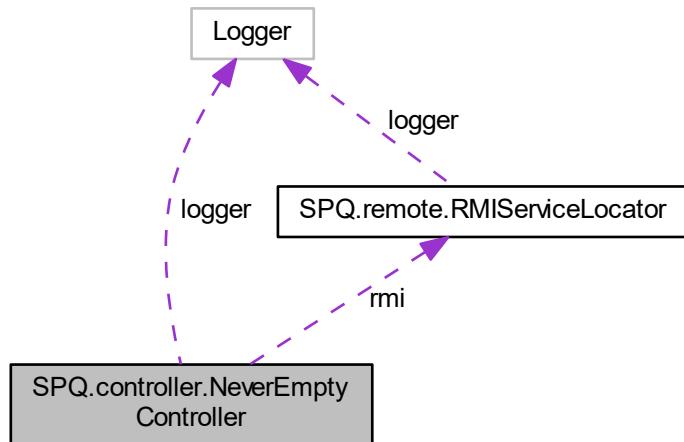
The documentation for this class was generated from the following file:

- NeverEmptyClient/src/main/java/SPQ/main/[Main.java](#)

## 7.26 SPQ.controller.NeverEmptyController Class Reference

This class acts as an intermediary between the graphic user interface and the classes that call the server. It mostly contains registration, login and payment methods.

Collaboration diagram for SPQ.controller.NeverEmptyController:



## Public Member Functions

- `NeverEmptyController (RMIServiceLocator rmi)`
- `UserDTO getUserDTO ()`
- `void setUserDTO (UserDTO userDTO)`
- `boolean registerGoogle (String username, String email, String password) throws RemoteException`
- `boolean registerFacebook (String username, String email, String password) throws RemoteException`
- `boolean registerNeverEmpty (String username, String email, String password) throws RemoteException`
- `boolean login (String username, String password) throws RemoteException`
- `ProductDTO getProducts () throws RemoteException`
- `UserDTO getUser (UserDTO userDTO) throws RemoteException`
- `boolean payWithPayPal (PaymentDTO paymentDTO) throws RemoteException`
- `boolean payWithVisa (PaymentDTO paymentDTO) throws RemoteException`
- `boolean updateUser (UserDTO userDTO) throws RemoteException`

### 7.26.1 Detailed Description

This class acts as an intermediary between the graphic user interface and the classes that call the server. It mostly contains registration, login and payment methods.

Definition at line 18 of file `NeverEmptyController.java`.

### 7.26.2 Constructor & Destructor Documentation

#### 7.26.2.1 NeverEmptyController()

```
SPQ.controller.NeverEmptyController.NeverEmptyController (
    RMIServiceLocator rmi )
```

Constructor of `NeverEmptyController` class.

**Parameters**

<i>rmi.</i>	Receives RMIServiceLocator instance.
-------------	--------------------------------------

Definition at line 27 of file NeverEmptyController.java.

### 7.26.3 Member Function Documentation

#### 7.26.3.1 getProducts()

`ProductDTO SPQ.controller.NeverEmptyController.getProducts ( ) throws RemoteException`

Asks to the [NeverEmptyServer](#) for the complete list of products.

**Returns**

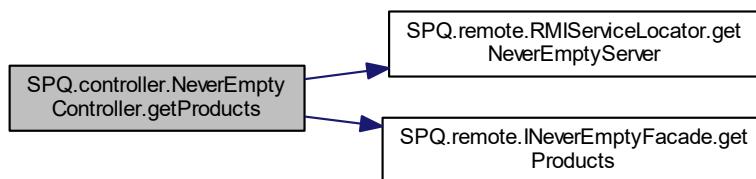
A ProductDTO that contains the full list of products.

**Exceptions**

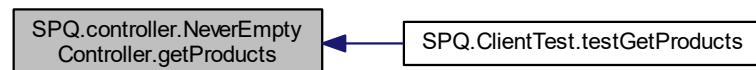
<i>RemoteException</i>	
------------------------	--

Definition at line 102 of file NeverEmptyController.java.

Here is the call graph for this function:



Here is the caller graph for this function:



### 7.26.3.2 getUser()

```
UserDTO SPQ.controller.NeverEmptyController.getUser (
    UserDTO userDTO ) throws RemoteException
```

Asks to the NeverEmpty server for all the data from the user specified in the given userDTO.

#### Parameters

<i>userDTO</i>	<input type="text"/>
----------------	----------------------

#### Returns

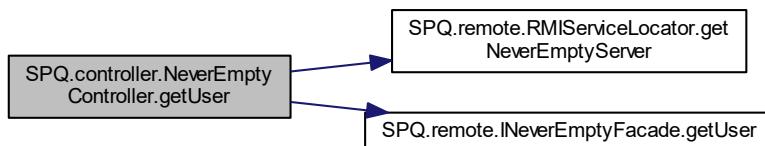
All the information of the user.

#### Exceptions

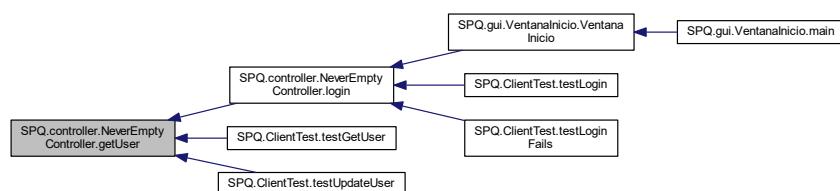
<i>RemoteException</i>	<input type="text"/>
------------------------	----------------------

Definition at line 111 of file NeverEmptyController.java.

Here is the call graph for this function:



Here is the caller graph for this function:



### 7.26.3.3 getUserDTO()

```
UserDTO SPQ.controller.NeverEmptyController.getUserDTO ( )
```

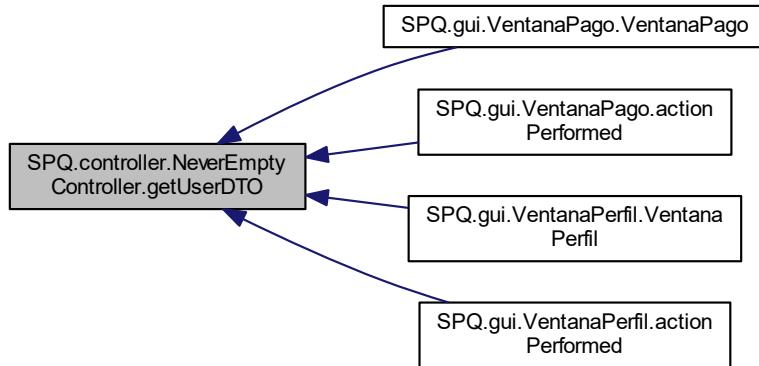
Returns the UserDTO attribute of the class.

#### Returns

UserDTO instance.

Definition at line 34 of file NeverEmptyController.java.

Here is the caller graph for this function:



### 7.26.3.4 login()

```
boolean SPQ.controller.NeverEmptyController.login (
    String username,
    String password ) throws RemoteException
```

Tries to login in NeverEmpty. It calls the login(UserDTO) function on NeverEmptyFacade and sets the NeverEmptyController UserDTO attribute with the received user.

#### Parameters

<i>username</i>	
<i>password</i>	

#### Returns

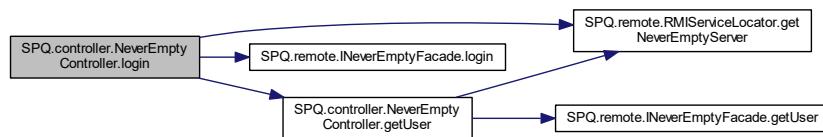
true if credentials are correct, false if not.

## Exceptions

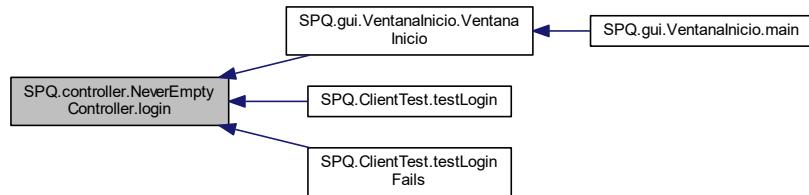
*RemoteException*

Definition at line 86 of file NeverEmptyController.java.

Here is the call graph for this function:



Here is the caller graph for this function:



### 7.26.3.5 payWithPayPal()

```
boolean SPQ.controller.NeverEmptyController.payWithPayPal (
    PaymentDTO paymentDTO ) throws RemoteException
```

Sends the payment information to the NeverEmpty server, who will manage the payment via PayPal.

#### Parameters

<i>paymentDTO</i> .	At least must be filled the paypalUsername, paypalPassword and total amount fields.
---------------------	---

#### Returns

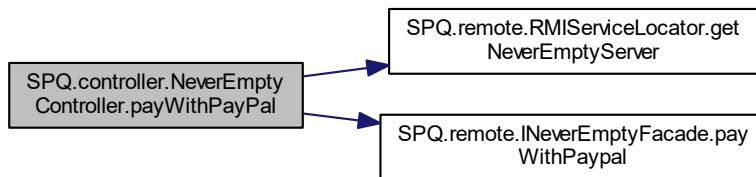
true if the payment has been successful, false if not.

### Exceptions

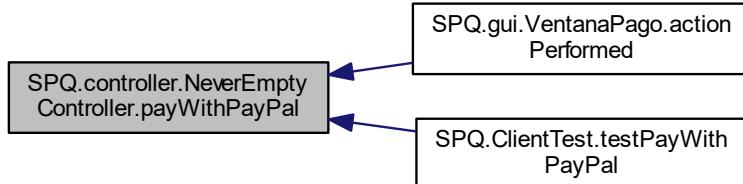
<i>RemoteException</i>	
------------------------	--

Definition at line 120 of file NeverEmptyController.java.

Here is the call graph for this function:



Here is the caller graph for this function:



#### 7.26.3.6 payWithVisa()

```

boolean SPQ.controller.NeverEmptyController.payWithVisa (
    PaymentDTO paymentDTO ) throws RemoteException
  
```

Sends the payment information to the NeverEmpty server, who will manage the payment via VISA.

### Parameters

<i>paymentDTO</i>	At least must be filled a cardnumber, cardholder, CVV, expDate and the amount to pay.
-------------------	---

**Returns**

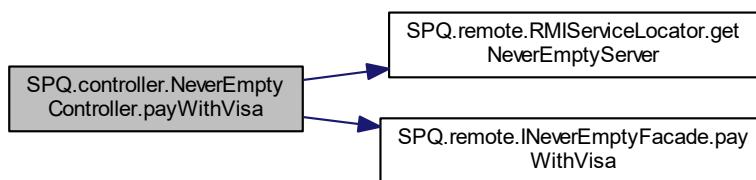
true if the payment has been successful, false if not.

**Exceptions**

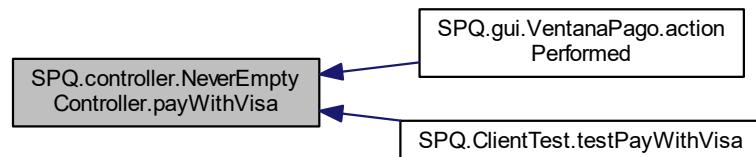
*RemoteException*

Definition at line 129 of file NeverEmptyController.java.

Here is the call graph for this function:



Here is the caller graph for this function:



### 7.26.3.7 registerFacebook()

```

boolean SPQ.controller.NeverEmptyController.registerFacebook (
    String username,
    String email,
    String password ) throws RemoteException

```

This method allows to sign up on NeverEmpty with an existing Facebook account. It calls the registerFacebook( $\leftarrow$  UserDTO) function of NeverEmptyFacade.

**Parameters**

<i>username.</i>	String, new NeverEmpty username.
<i>email.</i>	String, existing Facebook email.
<i>password.</i>	String, Facebook password.

**Returns**

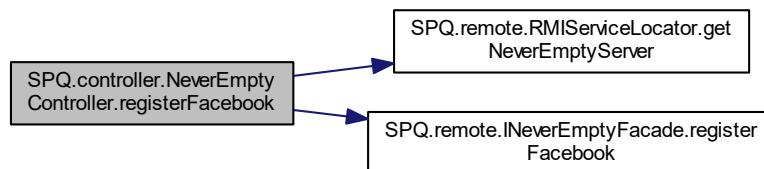
true if the credentials are correct. False if not.

**Exceptions**

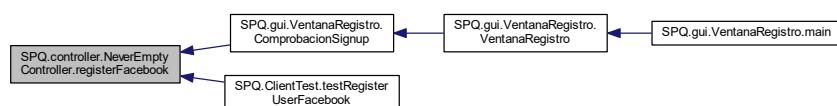
<i>RemoteException</i>	
------------------------	--

Definition at line 64 of file NeverEmptyController.java.

Here is the call graph for this function:



Here is the caller graph for this function:

**7.26.3.8 registerGoogle()**

```

boolean SPQ.controller.NeverEmptyController.registerGoogle (
    String username,
    String email,
    String password ) throws RemoteException
  
```

This method allows to sign up on NeverEmpty with an existing Google account. It calls the registerGoogle(UserDTO) function of NeverEmptyFacade.

### Parameters

<code>username.</code>	String, NeverEmpty username.
<code>email.</code>	String, existing Google email.
<code>password.</code>	String, Google password.

### Returns

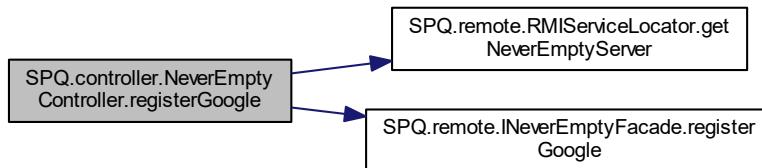
true if the credentials are correct. False if not.

### Exceptions

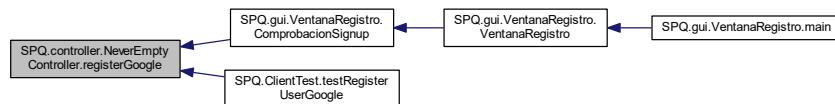
<code>RemoteException</code>	
------------------------------	--

Definition at line 52 of file NeverEmptyController.java.

Here is the call graph for this function:



Here is the caller graph for this function:



### 7.26.3.9 registerNeverEmpty()

```

boolean SPQ.controller.NeverEmptyController.registerNeverEmpty (
    String username,
    String email,
    String password ) throws RemoteException
  
```

Sign up on NeverEmpty. It calls registerNeverEmpty(UserDTO) function of NeverEmptyFacade.

### Parameters

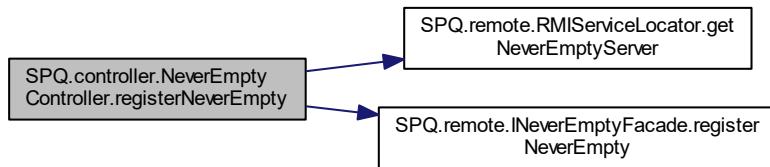
<code>username.</code>	String, new NeverEmpty username.
<code>email.</code>	String, existing email.
<code>password.</code>	String, new password for NeverEmpty.

### Exceptions

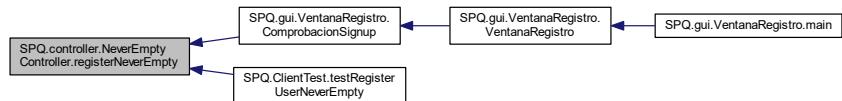
`RemoteException`

Definition at line 75 of file NeverEmptyController.java.

Here is the call graph for this function:



Here is the caller graph for this function:



#### 7.26.3.10 setUserDTO()

```
void SPQ.controller.NeverEmptyController.setUserDTO (
    UserDTO userDTO )
```

Sets UserDTO attribute of the class.

Definition at line 40 of file NeverEmptyController.java.

Here is the caller graph for this function:



#### 7.26.3.11 updateUser()

```
boolean SPQ.controller.NeverEmptyController.updateUser (
    UserDTO userDTO ) throws RemoteException
```

Sends user data to update the database. The email is the identifier of the user, so it is immutable.

##### Parameters

<i>userDTO</i>	
----------------	--

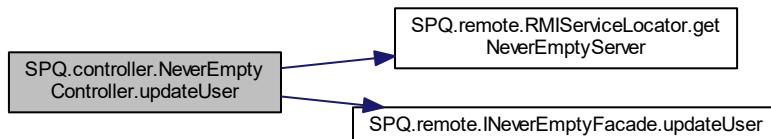
##### Returns

##### Exceptions

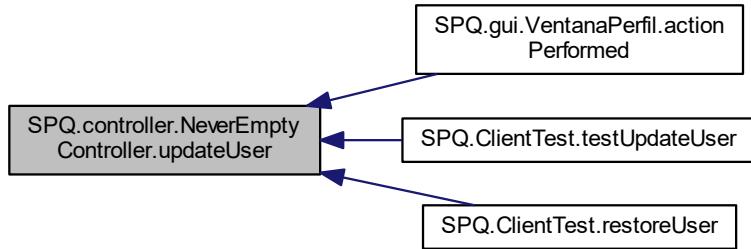
<i>RemoteException</i>	
------------------------	--

Definition at line 138 of file NeverEmptyController.java.

Here is the call graph for this function:



Here is the caller graph for this function:

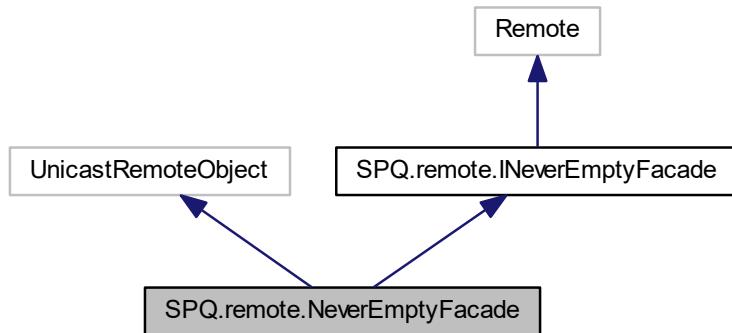


The documentation for this class was generated from the following file:

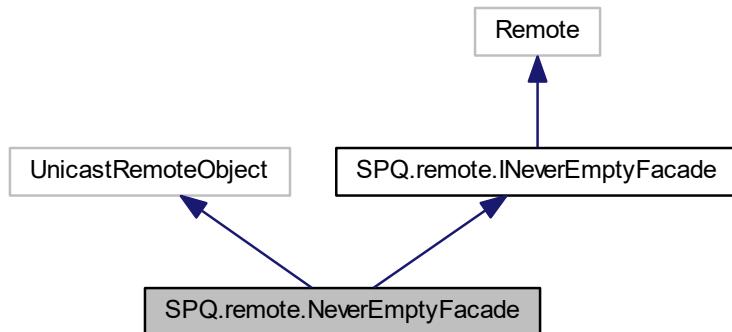
- `NeverEmptyClient/src/main/java/SPQ/controller/NeverEmptyController.java`

## 7.27 SPQ.remote.NeverEmptyFacade Class Reference

Inheritance diagram for `SPQ.remote.NeverEmptyFacade`:



Collaboration diagram for SPQ.remote.NeverEmptyFacade:



## Public Member Functions

- `NeverEmptyFacade (NeverEmptyServer neverEmptyServer) throws RemoteException`
- `boolean registerGoogle (UserDTO userDTO) throws RemoteException`
- `boolean registerFacebook (UserDTO userDTO) throws RemoteException`
- `boolean registerNeverEmpty (UserDTO userDTO) throws RemoteException`
- `boolean login (UserDTO user) throws RemoteException`
- `ProductDTO getProducts () throws RemoteException`
- `boolean updateUser (UserDTO userDTO) throws RemoteException`
- `UserDTO getUser (UserDTO user) throws RemoteException`
- `boolean payWithPaypal (PaymentDTO paymentDTO) throws RemoteException`
- `boolean payWithVisa (PaymentDTO paymentDTO) throws RemoteException`

### 7.27.1 Detailed Description

Definition at line 11 of file NeverEmptyFacade.java.

### 7.27.2 Constructor & Destructor Documentation

#### 7.27.2.1 NeverEmptyFacade()

```
SPQ.remote.NeverEmptyFacade.NeverEmptyFacade (
    NeverEmptyServer neverEmptyServer ) throws RemoteException
```

Definition at line 15 of file NeverEmptyFacade.java.

### 7.27.3 Member Function Documentation

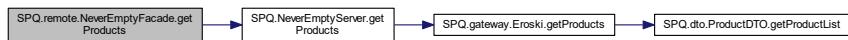
#### 7.27.3.1 getProducts()

`ProductDTO SPQ.remote.NeverEmptyFacade.getProducts ( ) throws RemoteException`

Implements [SPQ.remote.INeverEmptyFacade](#).

Definition at line 41 of file NeverEmptyFacade.java.

Here is the call graph for this function:



Here is the caller graph for this function:



#### 7.27.3.2 getUser()

`UserDTO SPQ.remote.NeverEmptyFacade.getUser ( UserDTO user ) throws RemoteException`

Implements [SPQ.remote.INeverEmptyFacade](#).

Definition at line 52 of file NeverEmptyFacade.java.

Here is the call graph for this function:



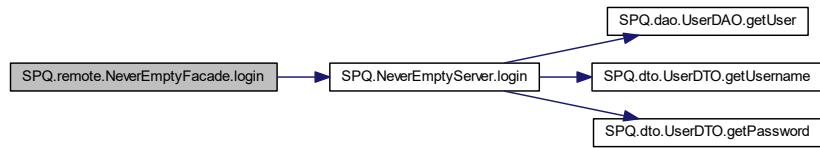
### 7.27.3.3 login()

```
boolean SPQ.remote.NeverEmptyFacade.login (
    UserDTO user ) throws RemoteException
```

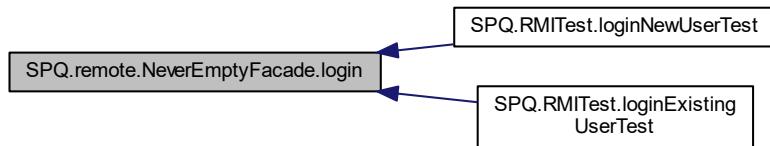
Implements [SPQ.remote.INeverEmptyFacade](#).

Definition at line 36 of file NeverEmptyFacade.java.

Here is the call graph for this function:



Here is the caller graph for this function:



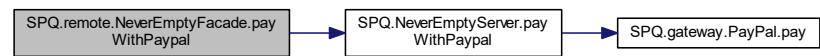
### 7.27.3.4 payWithPaypal()

```
boolean SPQ.remote.NeverEmptyFacade.payWithPaypal (
    PaymentDTO paymentDTO ) throws RemoteException
```

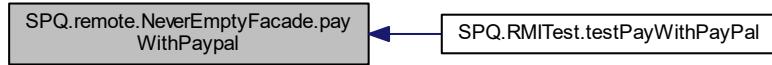
Implements [SPQ.remote.INeverEmptyFacade](#).

Definition at line 57 of file NeverEmptyFacade.java.

Here is the call graph for this function:



Here is the caller graph for this function:



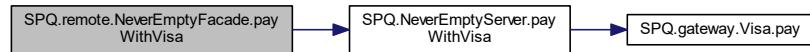
#### 7.27.3.5 payWithVisa()

```
boolean SPQ.remote.NeverEmptyFacade.payWithVisa (
    PaymentDTO paymentDTO ) throws RemoteException
```

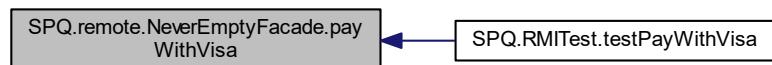
Implements [SPQ.remote.INeverEmptyFacade](#).

Definition at line 62 of file NeverEmptyFacade.java.

Here is the call graph for this function:



Here is the caller graph for this function:



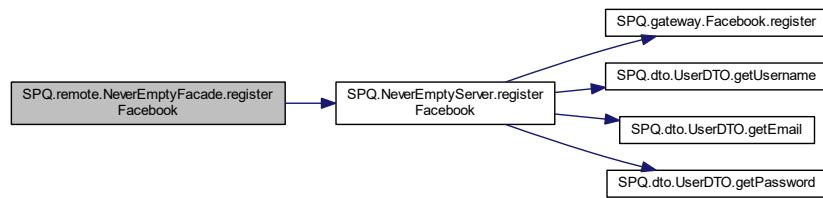
### 7.27.3.6 registerFacebook()

```
boolean SPQ.remote.NeverEmptyFacade.registerFacebook (
    UserDTO userDTO ) throws RemoteException
```

Implements [SPQ.remote.INeverEmptyFacade](#).

Definition at line 26 of file NeverEmptyFacade.java.

Here is the call graph for this function:



Here is the caller graph for this function:



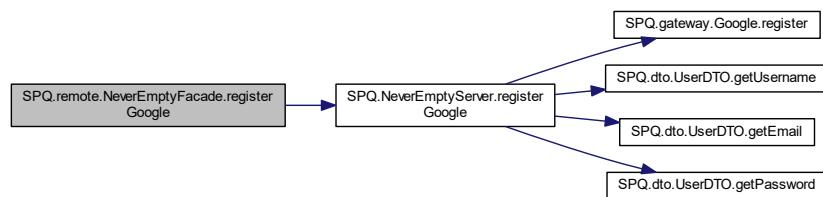
### 7.27.3.7 registerGoogle()

```
boolean SPQ.remote.NeverEmptyFacade.registerGoogle (
    UserDTO userDTO ) throws RemoteException
```

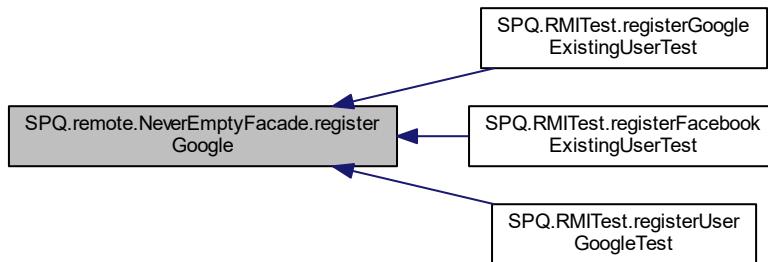
Implements [SPQ.remote.INeverEmptyFacade](#).

Definition at line 21 of file NeverEmptyFacade.java.

Here is the call graph for this function:



Here is the caller graph for this function:



#### 7.27.3.8 registerNeverEmpty()

```
boolean SPQ.remote.NeरेverEmptyFacade.registerNeverEmpty (
    UserDTO userDTO ) throws RemoteException
```

Implements [SPQ.remote.INeverEmptyFacade](#).

Definition at line 31 of file NeverEmptyFacade.java.

Here is the call graph for this function:



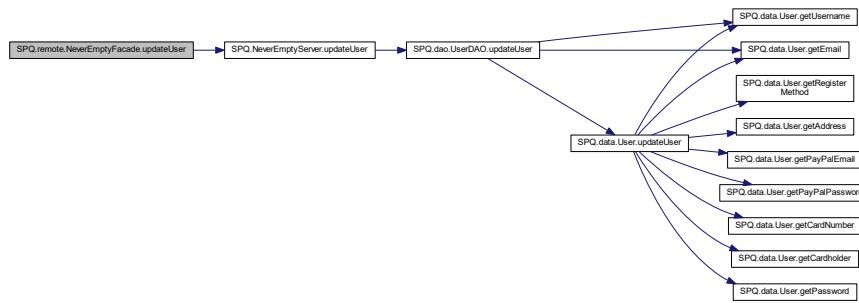
#### 7.27.3.9 updateUser()

```
boolean SPQ.remote.NeरеverEmptyFacade.updateUser (
    UserDTO userDTO ) throws RemoteException
```

Implements [SPQ.remote.INeverEmptyFacade](#).

Definition at line 46 of file NeverEmptyFacade.java.

Here is the call graph for this function:



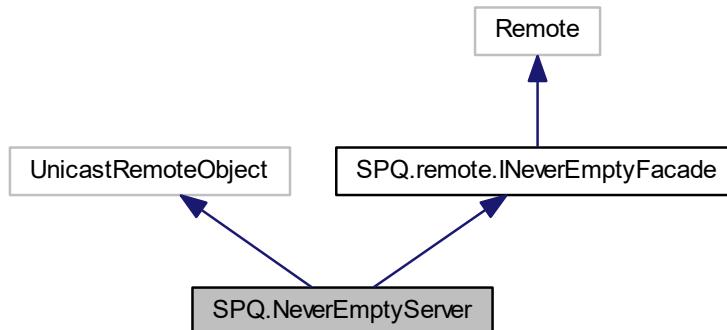
The documentation for this class was generated from the following file:

- NeverEmptyServer/src/main/java/SPQ/remote/[NeverEmptyFacade.java](#)

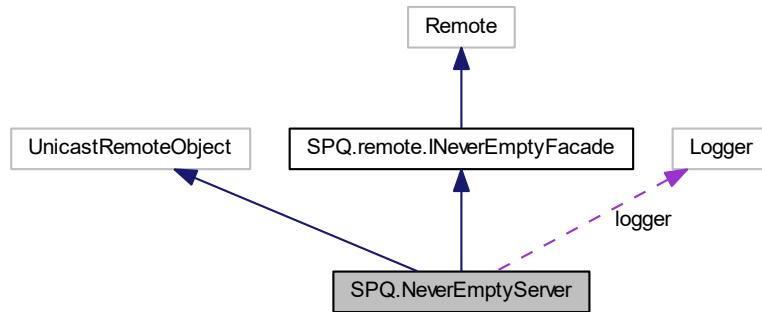
## 7.28 SPQ.NeverEmptyServer Class Reference

This class contains the methods that register and login a user, and recover information about them.

Inheritance diagram for SPQ.NeverEmptyServer:



Collaboration diagram for SPQ.NeverEmptyServer:



## Public Member Functions

- boolean `registerGoogle (UserDTO userDTO)`
- boolean `registerFacebook (UserDTO userDTO)`
- boolean `registerNeverEmpty (UserDTO userDTO) throws RemoteException`
- `UserDTO getUser (UserDTO user)`
- `boolean login (UserDTO user)`
- `ProductDTO getProducts ()`
- boolean `payWithPaypal (PaymentDTO paymentDTO)`
- boolean `payWithVisa (PaymentDTO paymentDTO)`
- boolean `updateUser (UserDTO userDTO)`

## Protected Member Functions

- `NeverEmptyServer () throws RemoteException`

### 7.28.1 Detailed Description

This class contains the methods that register and login a user, and recover information about them.

Definition at line 28 of file `NeverEmptyServer.java`.

### 7.28.2 Constructor & Destructor Documentation

#### 7.28.2.1 NeverEmptyServer()

```
SPQ.NeverEmptyServer.NeverEmptyServer ( ) throws RemoteException [protected]
```

Definition at line 30 of file `NeverEmptyServer.java`.

### 7.28.3 Member Function Documentation

#### 7.28.3.1 getProducts()

```
ProductDTO SPQ.NeverEmptyServer.getProducts ( )
```

This method ask to the neverEmpty server for the complete list of products

##### Returns

the complete list of products that the server contains

Implements [SPQ.remote.INeverEmptyFacade](#).

Definition at line 124 of file NeverEmptyServer.java.

Here is the call graph for this function:



Here is the caller graph for this function:



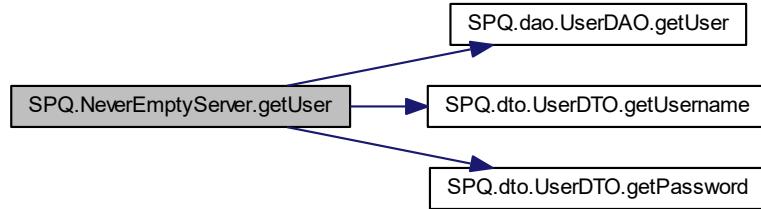
#### 7.28.3.2 getUser()

```
UserDTO SPQ.NeverEmptyServer.getUser (
    UserDTO user )
```

Implements [SPQ.remote.INeverEmptyFacade](#).

Definition at line 94 of file NeverEmptyServer.java.

Here is the call graph for this function:



Here is the caller graph for this function:



### 7.28.3.3 login()

```
boolean SPQ.NeverEmptyServer.login (
    UserDTO user )
```

This method login a user into server

#### Parameters

`UserDTO`

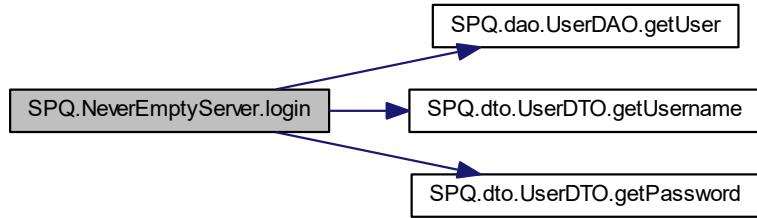
#### Returns

true if name and password coincide, false if not

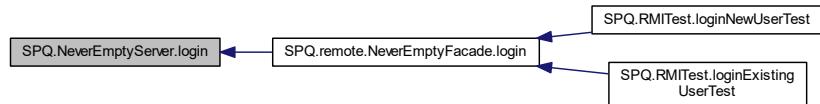
Implements [SPQ.remote.INeverEmptyFacade](#).

Definition at line 107 of file `NeverEmptyServer.java`.

Here is the call graph for this function:



Here is the caller graph for this function:



#### 7.28.3.4 payWithPaypal()

```
boolean SPQ.NeverEmptyServer.payWithPaypal (
    PaymentDTO paymentDTO )
```

This method allows a user to pay with his paypal account

##### Parameters

<code>paymentDTO</code>	
-------------------------	--

##### Returns

true if the payment connect correctly, false if not

Implements [SPQ.remote.INeverEmptyFacade](#).

Definition at line 143 of file `NeverEmptyServer.java`.

Here is the call graph for this function:



Here is the caller graph for this function:



#### 7.28.3.5 payWithVisa()

```
boolean SPQ.NeverEmptyServer.payWithVisa (
    PaymentDTO paymentDTO )
```

This method allows a user to pay with his visa account

##### Parameters

<i>paymentDTO</i>	<input type="text"/>
-------------------	----------------------

##### Returns

true if the payment connect correctly, false if not

Implements [SPQ.remote.INeverEmptyFacade](#).

Definition at line 161 of file NeverEmptyServer.java.

Here is the call graph for this function:



Here is the caller graph for this function:



#### 7.28.3.6 registerFacebook()

```
boolean SPQ.NeverEmptyServer.registerFacebook (
    UserDTO userDTO )
```

This methods allows register an user with his facebook account

##### Parameters

UserDTO	<input type="text"/>
---------	----------------------

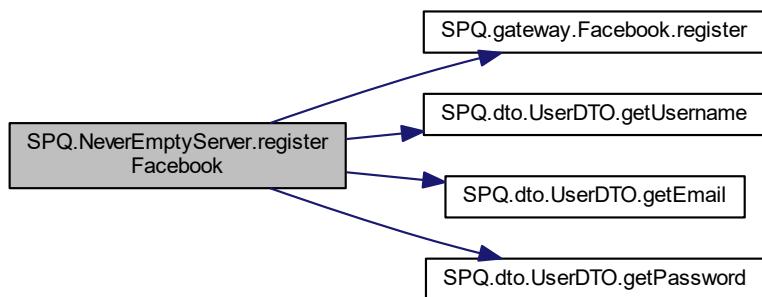
##### Returns

a user with username, email, password and the string "facebook"

Implements [SPQ.remote.INeverEmptyFacade](#).

Definition at line 61 of file NeverEmptyServer.java.

Here is the call graph for this function:



Here is the caller graph for this function:



### 7.28.3.7 registerGoogle()

```
boolean SPQ.NeverEmptyServer.registerGoogle (
    UserDTO userDTO )
```

This method allows to register an user with his google account

#### Parameters

UserDTO	
---------	--

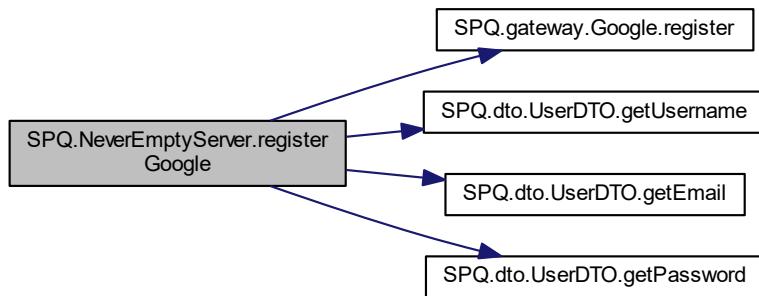
#### Returns

a user with username, email, password and the string "google"

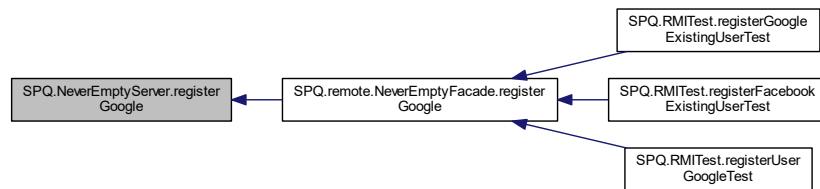
Implements [SPQ.remote.INeverEmptyFacade](#).

Definition at line 43 of file NeverEmptyServer.java.

Here is the call graph for this function:



Here is the caller graph for this function:



### 7.28.3.8 registerNeverEmpty()

```
boolean SPQ.NeverEmptyServer.registerNeverEmpty (
    UserDTO userDTO ) throws RemoteException
```

This method pick up all information about a user and register it into [NeverEmptyServer](#) moves the information to UserDAO to send it to server

#### Parameters

<i>UserDTO</i>	
----------------	--

#### Returns

if the information is send to server return true, if not return false

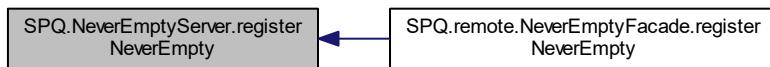
#### Exceptions

<i>remoteException</i>	
------------------------	--

Implements [SPQ.remote.INeverEmptyFacade](#).

Definition at line 82 of file NeverEmptyServer.java.

Here is the caller graph for this function:



### 7.28.3.9 updateUser()

```
boolean SPQ.NeverEmptyServer.updateUser (
    UserDTO userDTO )
```

This method allows a user to change their credentials except mail

#### Parameters

<i>UserDTO</i>	
<i>username</i>	
<i>password</i>	

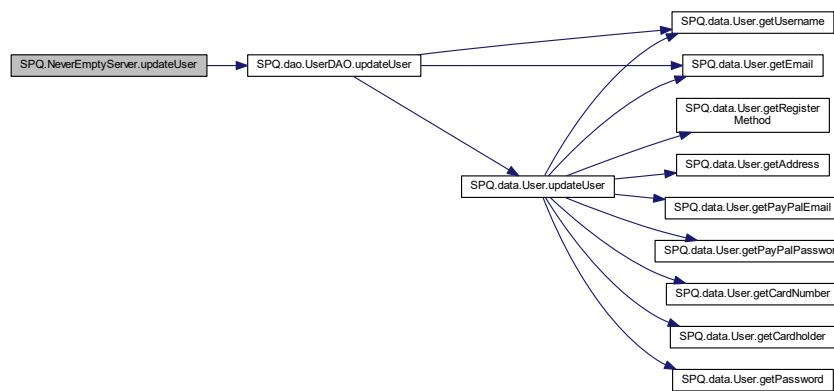
**Returns**

new user with the new credentials

Implements [SPQ.remote.INeverEmptyFacade](#).

Definition at line 180 of file `NeverEmptyServer.java`.

Here is the call graph for this function:



Here is the caller graph for this function:



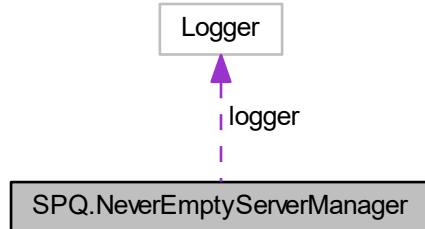
The documentation for this class was generated from the following file:

- [NeverEmptyServer/src/main/java/SPQ/NeverEmptyServer.java](#)

## 7.29 SPQ.NeverEmptyServerManager Class Reference

This class contains the main method to execute the server side.

Collaboration diagram for SPQ.NeverEmptyServerManager:



## Static Public Member Functions

- static void [main](#) (String[ ] args)

### 7.29.1 Detailed Description

This class contains the main method to execute the server side.

Definition at line 18 of file `NeverEmptyServerManager.java`.

### 7.29.2 Member Function Documentation

#### 7.29.2.1 `main()`

```
static void SPQ.NeverEmptyServerManager.main (
    String [ ] args ) [static]
```

Definition at line 22 of file `NeverEmptyServerManager.java`.

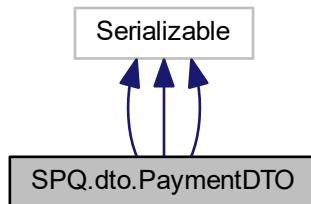
The documentation for this class was generated from the following file:

- [NeverEmptyServer/src/main/java/SPQ/NeverEmptyServerManager.java](#)

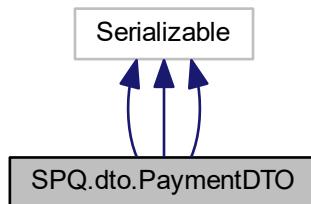
## 7.30 SPQ.dto.PaymentDTO Class Reference

This class prepares a payment to be transferred.

Inheritance diagram for SPQ.dto.PaymentDTO:



Collaboration diagram for SPQ.dto.PaymentDTO:



### Public Member Functions

- `PaymentDTO` (double total, long cardnumber, String cardholder, String expDate, int cvv)
- `PaymentDTO` (double total, String username, String password)
- double `getTotal` ()
- void `setTotal` (double total)
- long `getCardnumber` ()
- void `setCardnumber` (long cardnumber)
- String `getCardholder` ()
- void `setCardholder` (String cardholder)
- String `getExpDate` ()
- void `setExpDate` (String expDate)
- int `getCvv` ()
- void `setCvv` (int cvv)
- String `getUsername` ()
- void `setUsername` (String username)

- String `getPassword()`
- void `setPassword(String password)`
- `PaymentDTO(double total, long cardnumber, String cardholder, String expDate, int cvv)`
- `PaymentDTO(double total, String username, String password)`
- double `getTotal()`
- void `setTotal(double total)`
- long `getCardnumber()`
- void `setCardnumber(long cardnumber)`
- String `getCardholder()`
- void `setCardholder(String cardholder)`
- String `getExpDate()`
- void `setExpDate(String expDate)`
- int `getCvv()`
- void `setCvv(int cvv)`
- String `getUsername()`
- void `setUsername(String username)`
- String `getPassword()`
- void `setPassword(String password)`
- `PaymentDTO(double total, long cardnumber, String cardholder, String expDate, int cvv)`
- `PaymentDTO(int total, String username, String password)`
- double `getTotal()`
- void `setTotal(double total)`
- long `getCardnumber()`
- void `setCardnumber(long cardnumber)`
- String `getCardholder()`
- void `setCardholder(String cardholder)`
- String `getExpDate()`
- void `setExpDate(String expDate)`
- int `getCvv()`
- void `setCvv(int cvv)`
- String `getUsername()`
- void `setUsername(String username)`
- String `getPassword()`
- void `setPassword(String password)`

### 7.30.1 Detailed Description

This class prepares a payment to be transferred.

Definition at line 12 of file PaymentDTO.java.

### 7.30.2 Constructor & Destructor Documentation

#### 7.30.2.1 PaymentDTO() [1/6]

```
SPQ.dto.PaymentDTO.PaymentDTO (
    double total,
    long cardnumber,
    String cardholder,
    String expDate,
    int cvv )
```

Constructor of paymentDTO.class

**Parameters**

<i>total</i>	
<i>cardnumber</i>	
<i>cardholder</i>	
<i>expDate</i>	
<i>cvv</i>	

Definition at line 33 of file PaymentDTO.java.

**7.30.2.2 PaymentDTO() [2/6]**

```
SPQ.dto.PaymentDTO.PaymentDTO (
    double total,
    String username,
    String password )
```

Contructor of paymentDTO.class

**Parameters**

<i>total</i>	
<i>username</i>	
<i>password</i>	

Definition at line 48 of file PaymentDTO.java.

**7.30.2.3 PaymentDTO() [3/6]**

```
SPQ.dto.PaymentDTO.PaymentDTO (
    double total,
    long cardnumber,
    String cardholder,
    String expDate,
    int cvv )
```

Definition at line 19 of file PaymentDTO.java.

**7.30.2.4 PaymentDTO() [4/6]**

```
SPQ.dto.PaymentDTO.PaymentDTO (
    double total,
    String username,
    String password )
```

Definition at line 27 of file PaymentDTO.java.

**7.30.2.5 PaymentDTO() [5/6]**

```
SPQ.dto.PaymentDTO.PaymentDTO (
    double total,
    long cardnumber,
    String cardholder,
    String expDate,
    int cvv )
```

Definition at line 19 of file PaymentDTO.java.

**7.30.2.6 PaymentDTO() [6/6]**

```
SPQ.dto.PaymentDTO.PaymentDTO (
    int total,
    String username,
    String password )
```

Definition at line 27 of file PaymentDTO.java.

### 7.30.3 Member Function Documentation

**7.30.3.1 getCardholder() [1/3]**

```
String SPQ.dto.PaymentDTO.getCardholder ( )
```

Definition at line 47 of file PaymentDTO.java.

**7.30.3.2 getCardholder() [2/3]**

```
String SPQ.dto.PaymentDTO.getCardholder ( )
```

Definition at line 47 of file PaymentDTO.java.

### 7.30.3.3 getCardholder() [3/3]

```
String SPQ.dto.PaymentDTO.getCardholder ( )
```

Returns the cardholder attribute of the class.

**Returns**

cardholder instance.

Definition at line 89 of file PaymentDTO.java.

Here is the caller graph for this function:



### 7.30.3.4 getCardnumber() [1/3]

```
long SPQ.dto.PaymentDTO.getCardnumber ( )
```

Definition at line 41 of file PaymentDTO.java.

### 7.30.3.5 getCardnumber() [2/3]

```
long SPQ.dto.PaymentDTO.getCardnumber ( )
```

Definition at line 41 of file PaymentDTO.java.

### 7.30.3.6 getCardnumber() [3/3]

```
long SPQ.dto.PaymentDTO.getCardnumber ( )
```

Returns the cardnumber attribute of the class.

**Returns**

cardnumber instance.

Definition at line 74 of file PaymentDTO.java.

Here is the caller graph for this function:



**7.30.3.7 getCVV() [1/3]**

```
int SPQ.dto.PaymentDTO.getCVV( )
```

Definition at line 59 of file PaymentDTO.java.

**7.30.3.8 getCVV() [2/3]**

```
int SPQ.dto.PaymentDTO.getCVV( )
```

Definition at line 59 of file PaymentDTO.java.

**7.30.3.9 getCVV() [3/3]**

```
int SPQ.dto.PaymentDTO.getCVV( )
```

Returns the cvv attribute of the class.

**Returns**

cvv instance.

Definition at line 119 of file PaymentDTO.java.

Here is the caller graph for this function:

**7.30.3.10 getExpDate() [1/3]**

```
String SPQ.dto.PaymentDTO.getExpDate( )
```

Definition at line 53 of file PaymentDTO.java.

**7.30.3.11 getExpDate() [2/3]**

```
String SPQ.dto.PaymentDTO.getExpDate ( )
```

Definition at line 53 of file PaymentDTO.java.

**7.30.3.12 getExpDate() [3/3]**

```
String SPQ.dto.PaymentDTO.getExpDate ( )
```

Returns the expDaate attribute of the class.

**Returns**

ExpDate instance.

Definition at line 104 of file PaymentDTO.java.

Here is the caller graph for this function:

**7.30.3.13 getPassword() [1/3]**

```
String SPQ.dto.PaymentDTO.getPassword ( )
```

Definition at line 71 of file PaymentDTO.java.

**7.30.3.14 getPassword() [2/3]**

```
String SPQ.dto.PaymentDTO.getPassword ( )
```

Definition at line 71 of file PaymentDTO.java.

### 7.30.3.15 getPassword() [3/3]

```
String SPQ.dto.PaymentDTO.getPassword( )
```

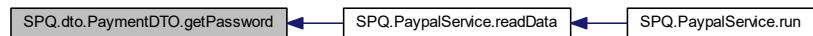
Returns the [UserDTO](#) attribute of the class.

**Returns**

[UserDTO](#) instance.

Definition at line 149 of file PaymentDTO.java.

Here is the caller graph for this function:



### 7.30.3.16 getTotal() [1/3]

```
double SPQ.dto.PaymentDTO.getTotal( )
```

Definition at line 35 of file PaymentDTO.java.

### 7.30.3.17 getTotal() [2/3]

```
double SPQ.dto.PaymentDTO.getTotal( )
```

Definition at line 35 of file PaymentDTO.java.

### 7.30.3.18 getTotal() [3/3]

```
double SPQ.dto.PaymentDTO.getTotal( )
```

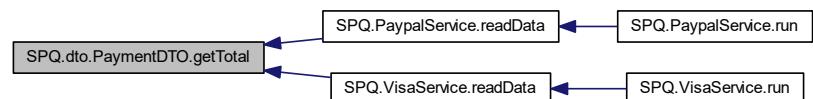
Returns the total attribute of the class.

**Returns**

total instance.

Definition at line 59 of file PaymentDTO.java.

Here is the caller graph for this function:



**7.30.3.19 getUsername() [1/3]**

```
String SPQ.dto.PaymentDTO.getUsername( )
```

Definition at line 65 of file PaymentDTO.java.

**7.30.3.20 getUsername() [2/3]**

```
String SPQ.dto.PaymentDTO.getUsername( )
```

Definition at line 65 of file PaymentDTO.java.

**7.30.3.21 getUsername() [3/3]**

```
String SPQ.dto.PaymentDTO.getUsername( )
```

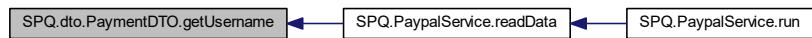
Returns the [UserDTO](#) attribute of the class.

**Returns**

[UserDTO](#) instance.

Definition at line 134 of file PaymentDTO.java.

Here is the caller graph for this function:

**7.30.3.22 setCardholder() [1/3]**

```
void SPQ.dto.PaymentDTO.setCardholder( String cardholder )
```

Definition at line 50 of file PaymentDTO.java.

**7.30.3.23 setCardholder() [2/3]**

```
void SPQ.dto.PaymentDTO.setCardholder (
    String cardholder )
```

Definition at line 50 of file PaymentDTO.java.

**7.30.3.24 setCardholder() [3/3]**

```
void SPQ.dto.PaymentDTO.setCardholder (
    String cardholder )
```

Sets cardHolder attribute of the class.

Definition at line 96 of file PaymentDTO.java.

**7.30.3.25 setCardnumber() [1/3]**

```
void SPQ.dto.PaymentDTO.setCardnumber (
    long cardnumber )
```

Definition at line 44 of file PaymentDTO.java.

**7.30.3.26 setCardnumber() [2/3]**

```
void SPQ.dto.PaymentDTO.setCardnumber (
    long cardnumber )
```

Definition at line 44 of file PaymentDTO.java.

**7.30.3.27 setCardnumber() [3/3]**

```
void SPQ.dto.PaymentDTO.setCardnumber (
    long cardnumber )
```

Sets cardNumber attribute of the class.

Definition at line 81 of file PaymentDTO.java.

**7.30.3.28 setCvv() [1/3]**

```
void SPQ.dto.PaymentDTO.setCvv (
    int cvv )
```

Definition at line 62 of file PaymentDTO.java.

**7.30.3.29 setCvv() [2/3]**

```
void SPQ.dto.PaymentDTO.setCvv (
    int cvv )
```

Definition at line 62 of file PaymentDTO.java.

**7.30.3.30 setCvv() [3/3]**

```
void SPQ.dto.PaymentDTO.setCvv (
    int cvv )
```

Sets cvv attribute of the class.

Definition at line 126 of file PaymentDTO.java.

**7.30.3.31 setExpDate() [1/3]**

```
void SPQ.dto.PaymentDTO.setExpDate (
    String expDate )
```

Definition at line 56 of file PaymentDTO.java.

**7.30.3.32 setExpDate() [2/3]**

```
void SPQ.dto.PaymentDTO.setExpDate (
    String expDate )
```

Definition at line 56 of file PaymentDTO.java.

**7.30.3.33 setExpDate()** [3/3]

```
void SPQ.dto.PaymentDTO.setExpDate (
    String expDate )
```

Sets expDate attribute of the class.

Definition at line 111 of file PaymentDTO.java.

**7.30.3.34 setPassword()** [1/3]

```
void SPQ.dto.PaymentDTO.setPassword (
    String password )
```

Definition at line 74 of file PaymentDTO.java.

**7.30.3.35 setPassword()** [2/3]

```
void SPQ.dto.PaymentDTO.setPassword (
    String password )
```

Definition at line 74 of file PaymentDTO.java.

**7.30.3.36 setPassword()** [3/3]

```
void SPQ.dto.PaymentDTO.setPassword (
    String password )
```

Sets password attribute of the class.

Definition at line 156 of file PaymentDTO.java.

**7.30.3.37 setTotal()** [1/3]

```
void SPQ.dto.PaymentDTO.setTotal (
    double total )
```

Definition at line 38 of file PaymentDTO.java.

**7.30.3.38 setTotal() [2/3]**

```
void SPQ.dto.PaymentDTO.setTotal (
    double total )
```

Definition at line 38 of file PaymentDTO.java.

**7.30.3.39 setTotal() [3/3]**

```
void SPQ.dto.PaymentDTO.setTotal (
    double total )
```

Sets total attribute of the class.

Definition at line 66 of file PaymentDTO.java.

**7.30.3.40 setUsername() [1/3]**

```
void SPQ.dto.PaymentDTO.setUsername (
    String username )
```

Definition at line 68 of file PaymentDTO.java.

**7.30.3.41 setUsername() [2/3]**

```
void SPQ.dto.PaymentDTO.setUsername (
    String username )
```

Definition at line 68 of file PaymentDTO.java.

**7.30.3.42 setUsername() [3/3]**

```
void SPQ.dto.PaymentDTO.setUsername (
    String username )
```

Sets username attribute of the class.

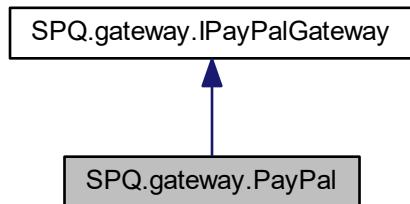
Definition at line 141 of file PaymentDTO.java.

The documentation for this class was generated from the following file:

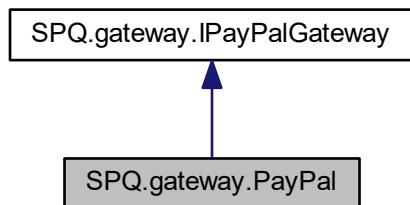
- NeverEmptyServer/src/main/java/SPQ/dto/[PaymentDTO.java](#)

## 7.31 SPQ.gateway.PayPal Class Reference

Inheritance diagram for SPQ.gateway.PayPal:



Collaboration diagram for SPQ.gateway.PayPal:



### Public Member Functions

- [PayPal](#) (String serverIP, String serverPort)
- boolean [pay](#) ([PaymentDTO](#) paymentDTO)

#### 7.31.1 Detailed Description

Definition at line 14 of file [PayPal.java](#).

#### 7.31.2 Constructor & Destructor Documentation

### 7.31.2.1 PayPal()

```
SPQ.gateway.PayPal.PayPal (
    String serverIP,
    String serverPort )
```

Definition at line 20 of file `PayPal.java`.

### 7.31.3 Member Function Documentation

#### 7.31.3.1 pay()

```
boolean SPQ.gateway.PayPal.pay (
    PaymentDTO paymentDTO )
```

Implements [SPQ.gateway.IPayPalGateway](#).

Definition at line 31 of file `PayPal.java`.

Here is the caller graph for this function:

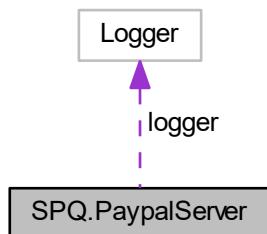


The documentation for this class was generated from the following file:

- [NeverEmptyServer/src/main/java/SPQ/gateway/PayPal.java](#)

## 7.32 SPQ.PaypalServer Class Reference

Collaboration diagram for SPQ.PaypalServer:



## Static Public Member Functions

- static void [main](#) (String args[ ])

### 7.32.1 Detailed Description

Definition at line 8 of file [PaypalServer.java](#).

### 7.32.2 Member Function Documentation

#### 7.32.2.1 main()

```
static void SPQ.PaypalServer.main (
    String args[ ] ) [static]
```

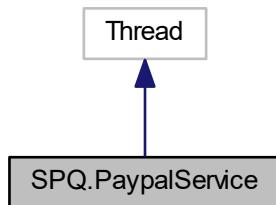
Definition at line 13 of file [PaypalServer.java](#).

The documentation for this class was generated from the following file:

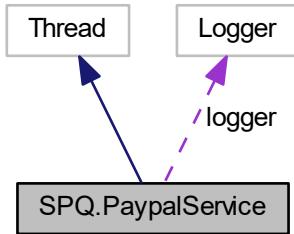
- [PaypalService/src/main/java/SPQ/PaypalServer.java](#)

## 7.33 SPQ.PaypalService Class Reference

Inheritance diagram for SPQ.PaypalService:



Collaboration diagram for SPQ.PaypalService:



## Public Member Functions

- [PaypalService](#) (Socket socket)
- void [run](#) ()
- String [readData](#) (PaymentDTO paymentDTO)

### 7.33.1 Detailed Description

Definition at line 15 of file PaypalService.java.

### 7.33.2 Constructor & Destructor Documentation

#### 7.33.2.1 PaypalService()

```
SPQ.PaypalService.PaypalService (
    Socket socket )
```

Definition at line 20 of file PaypalService.java.

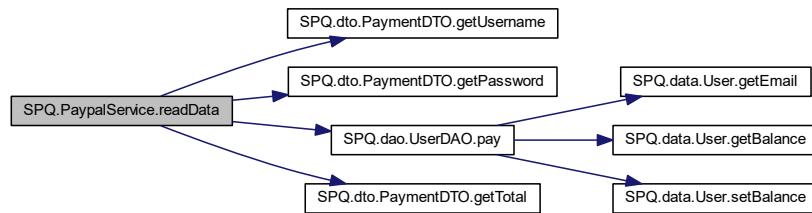
### 7.33.3 Member Function Documentation

### 7.33.3.1 `readData()`

```
String SPQ.PaypalService.readData (
    PaymentDTO paymentDTO )
```

Definition at line 53 of file `PaypalService.java`.

Here is the call graph for this function:



Here is the caller graph for this function:

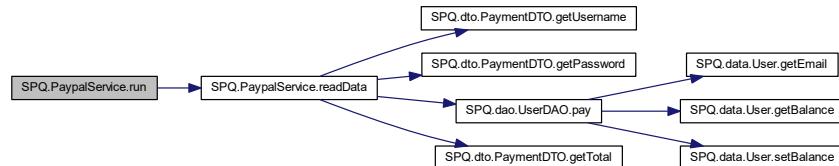


### 7.33.3.2 `run()`

```
void SPQ.PaypalService.run ( )
```

Definition at line 31 of file `PaypalService.java`.

Here is the call graph for this function:

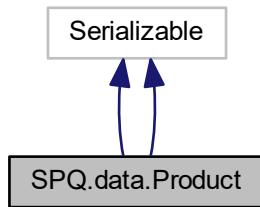


The documentation for this class was generated from the following file:

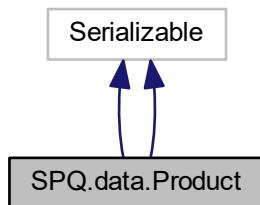
- `PaypalService/src/main/java/SPQ/PaypalService.java`

## 7.34 SPQ.data.Product Class Reference

Inheritance diagram for SPQ.data.Product:



Collaboration diagram for SPQ.data.Product:



### Public Member Functions

- `Product (String name, double price)`
- `Product (String name, double price, double sale)`
- `double getSale ()`
- `void setSale (double sale)`
- `String getName ()`
- `void setName (String name)`
- `double getPrice ()`
- `void setPrice (double price)`
- `Product (String name, double price, int quantity, double sale)`
- `String getName ()`
- `void setName (String name)`
- `double getPrice ()`
- `void setPrice (double price)`
- `int getQuantity ()`
- `void setQuantity (int quantity)`
- `double getSale ()`
- `void setSale (double sale)`

### 7.34.1 Detailed Description

Definition at line 11 of file Product.java.

### 7.34.2 Constructor & Destructor Documentation

#### 7.34.2.1 Product() [1/3]

```
SPQ.data.Product.Product (
    String name,
    double price )
```

Definition at line 18 of file Product.java.

#### 7.34.2.2 Product() [2/3]

```
SPQ.data.Product.Product (
    String name,
    double price,
    double sale )
```

Definition at line 23 of file Product.java.

#### 7.34.2.3 Product() [3/3]

```
SPQ.data.Product.Product (
    String name,
    double price,
    int quantity,
    double sale )
```

Constructor of Product.class

##### Parameters

<i>name</i>	
<i>price</i>	
<i>quantity</i>	
<i>sale</i>	

Definition at line 30 of file Product.java.

### 7.34.3 Member Function Documentation

#### 7.34.3.1 getName() [1/2]

```
String SPQ.data.Product.getName ( )
```

Definition at line 38 of file Product.java.

Here is the caller graph for this function:



#### 7.34.3.2 getName() [2/2]

```
String SPQ.data.Product.getName ( )
```

Return the name attribute of the class

##### Returns

name instance

Definition at line 42 of file Product.java.

#### 7.34.3.3 getPrice() [1/2]

```
double SPQ.data.Product.getPrice ( )
```

Definition at line 44 of file Product.java.

Here is the caller graph for this function:



#### 7.34.3.4 getPrice() [2/2]

```
double SPQ.data.Product.getPrice ( )
```

Return the price attribute of the class

##### Returns

price instance

Definition at line 57 of file Product.java.

#### 7.34.3.5 getQuantity()

```
int SPQ.data.Product.getQuantity ( )
```

Return the quantity attribute of the class

##### Returns

quantity instance

Definition at line 73 of file Product.java.

Here is the caller graph for this function:

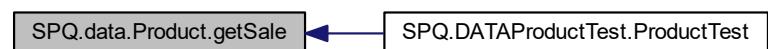


#### 7.34.3.6 getSale() [1/2]

```
double SPQ.data.Product.getSale ( )
```

Definition at line 30 of file Product.java.

Here is the caller graph for this function:



**7.34.3.7 getSale() [2/2]**

```
double SPQ.data.Product.getSale ( )
```

Return the sale attribute of the class

**Returns**

sale instance

Definition at line 89 of file Product.java.

**7.34.3.8 setName() [1/2]**

```
void SPQ.data.Product.setName (
    String name )
```

Definition at line 41 of file Product.java.

Here is the caller graph for this function:

**7.34.3.9 setName() [2/2]**

```
void SPQ.data.Product.setName (
    String name )
```

Sets name attribute of the class.

Definition at line 49 of file Product.java.

**7.34.3.10 setPrice() [1/2]**

```
void SPQ.data.Product.setPrice (
    double price )
```

Definition at line 47 of file Product.java.

Here is the caller graph for this function:

**7.34.3.11 setPrice() [2/2]**

```
void SPQ.data.Product.setPrice (
    double price )
```

Sets price attribute of the class.

Definition at line 64 of file Product.java.

**7.34.3.12 setQuantity()**

```
void SPQ.data.Product.setQuantity (
    int quantity )
```

Sets quantity attribute of the class.

Definition at line 80 of file Product.java.

Here is the caller graph for this function:

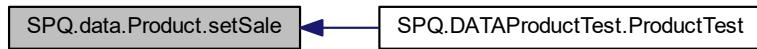


## 7.34.3.13 setSale() [1/2]

```
void SPQ.data.Product.setSale (
    double sale )
```

Definition at line 34 of file Product.java.

Here is the caller graph for this function:



## 7.34.3.14 setSale() [2/2]

```
void SPQ.data.Product.setSale (
    double sale )
```

Sets sale attribute of the class.

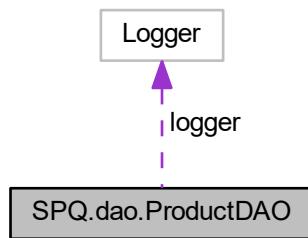
Definition at line 96 of file Product.java.

The documentation for this class was generated from the following file:

- EroskiService/src/main/java/SPQ/data/Product.java

## 7.35 SPQ.dao.ProductDAO Class Reference

Collaboration diagram for SPQ.dao.ProductDAO:



## Public Member Functions

- `ProductDAO ()`
- `void storeProduct (Product product)`
- `ArrayList< Product > getProducts ()`
- `List< Product > deleteProduct ()`

### 7.35.1 Detailed Description

Definition at line 16 of file ProductDAO.java.

### 7.35.2 Constructor & Destructor Documentation

#### 7.35.2.1 ProductDAO()

```
SPQ.dao.ProductDAO.ProductDAO ( )
```

Definition at line 21 of file ProductDAO.java.

### 7.35.3 Member Function Documentation

#### 7.35.3.1 deleteProduct()

```
List<Product> SPQ.dao.ProductDAO.deleteProduct ( )
```

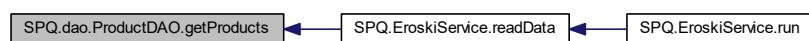
Definition at line 78 of file ProductDAO.java.

#### 7.35.3.2 getProducts()

```
ArrayList<Product> SPQ.dao.ProductDAO.getProducts ( )
```

Definition at line 47 of file ProductDAO.java.

Here is the caller graph for this function:



### 7.35.3.3 storeProduct()

```
void SPQ.dao.ProductDAO.storeProduct (
    Product product )
```

Definition at line 25 of file ProductDAO.java.

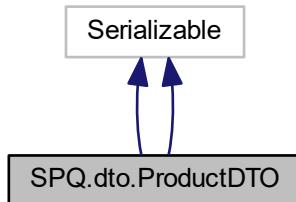
The documentation for this class was generated from the following file:

- EroskiService/src/main/java/SPQ/dao/ProductDAO.java

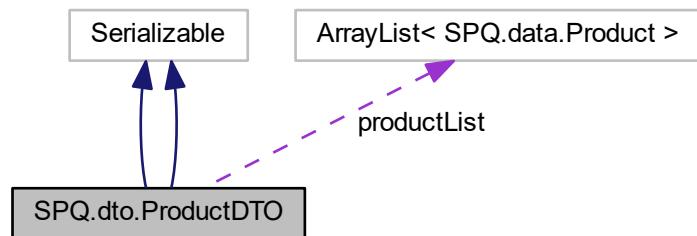
## 7.36 SPQ.dto.ProductDTO Class Reference

This class prepares a product to be transferred.

Inheritance diagram for SPQ.dto.ProductDTO:



Collaboration diagram for SPQ.dto.ProductDTO:



## Public Member Functions

- `ArrayList< Product > getProductList ()`
- `void setProductList (ArrayList< Product > productList)`
- `ProductDTO (ArrayList< Product > productList)`
- `ArrayList< Product > getProductList ()`
- `void setProductList (ArrayList< Product > productList)`
- `ProductDTO (ArrayList< Product > productList)`

### 7.36.1 Detailed Description

This class prepares a product to be transferred.

Definition at line 8 of file ProductDTO.java.

### 7.36.2 Constructor & Destructor Documentation

#### 7.36.2.1 ProductDTO() [1/2]

```
SPQ.dto.ProductDTO.ProductDTO (
    ArrayList< Product > productList )
```

Definition at line 22 of file ProductDTO.java.

#### 7.36.2.2 ProductDTO() [2/2]

```
SPQ.dto.ProductDTO.ProductDTO (
    ArrayList< Product > productList )
```

Constructor of `ProductDTO`

#### Parameters

<code>productList</code>	<input type="button" value=""/>
--------------------------	---------------------------------

Definition at line 43 of file ProductDTO.java.

### 7.36.3 Member Function Documentation

### 7.36.3.1 get productList() [1/2]

```
ArrayList<Product> SPQ.dto.ProductDTO.getProductList ( )
```

Definition at line 14 of file ProductDTO.java.

Here is the caller graph for this function:



### 7.36.3.2 get productList() [2/2]

```
ArrayList<Product> SPQ.dto.ProductDTO.getProductList ( )
```

This method returns the complete list of products in the list

#### Returns

productList

Definition at line 27 of file ProductDTO.java.

### 7.36.3.3 set productList() [1/2]

```
void SPQ.dto.ProductDTO.setProductList (
    ArrayList< Product > productList )
```

Definition at line 18 of file ProductDTO.java.

### 7.36.3.4 set productList() [2/2]

```
void SPQ.dto.ProductDTO.setProductList (
    ArrayList< Product > productList )
```

Sets productList attribute of the class.

Definition at line 34 of file ProductDTO.java.

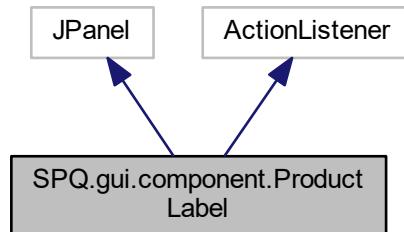
The documentation for this class was generated from the following file:

- EroskiService/src/main/java/SPQ/dto/ProductDTO.java

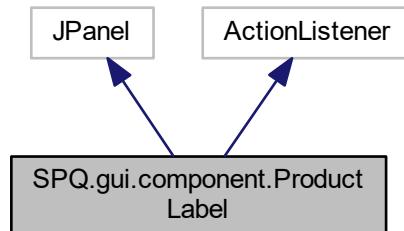
## 7.37 SPQ.gui.component.ProductLabel Class Reference

This component extends JPanel. Represents one product. [ProductLabel](#) consists of four labels: product name, price, sale (if > 0%), quantity; and two buttons: to increase the quantity by one and to subtract by one.

Inheritance diagram for SPQ.gui.component.ProductLabel:



Collaboration diagram for SPQ.gui.component.ProductLabel:



### Public Member Functions

- `ProductLabel (String productName, String price, String sale, VentanaPrincipal vp)`
- `ProductLabel (String productName, String price, String quantity, String sale, VentanaPrincipal vp)`
- `void actionPerformed (ActionEvent e)`
- `JLabel getSale ()`
- `void setSale (JLabel sale)`
- `JLabel getproductName ()`
- `void setproductName (JLabel productName)`
- `JLabel getPrice ()`
- `void setPrice (JLabel price)`
- `JLabel getQuantity ()`
- `void setQuantity (JLabel quantity)`
- `JButton getbAdd ()`
- `void setbAdd (JButton bAdd)`
- `JButton getbSub ()`
- `void setbSub (JButton bSub)`

### 7.37.1 Detailed Description

This component extends JPanel. Represents one product. `ProductLabel` consists of four labels: product name, price, sale (if > 0%), quantity; and two buttons: to increase the quantity by one and to subtract by one.

Definition at line 31 of file ProductLabel.java.

### 7.37.2 Constructor & Destructor Documentation

#### 7.37.2.1 `ProductLabel()` [1/2]

```
SPQ.gui.component.ProductLabel.ProductLabel (
    String productName,
    String price,
    String sale,
    VentanaPrincipal vp )
```

Definition at line 42 of file ProductLabel.java.

#### 7.37.2.2 `ProductLabel()` [2/2]

```
SPQ.gui.component.ProductLabel.ProductLabel (
    String productName,
    String price,
    String quantity,
    String sale,
    VentanaPrincipal vp )
```

Definition at line 56 of file ProductLabel.java.

### 7.37.3 Member Function Documentation

#### 7.37.3.1 `actionPerformed()`

```
void SPQ.gui.component.ProductLabel.actionPerformed (
    ActionEvent e )
```

Definition at line 139 of file ProductLabel.java.

#### 7.37.3.2 getbAdd()

```
JButton SPQ.gui.component.ProductLabel.getbAdd ( )
```

Definition at line 215 of file ProductLabel.java.

#### 7.37.3.3 getbSub()

```
JButton SPQ.gui.component.ProductLabel.getbSub ( )
```

Definition at line 223 of file ProductLabel.java.

#### 7.37.3.4 getPrice()

```
JLabel SPQ.gui.component.ProductLabel.getPrice ( )
```

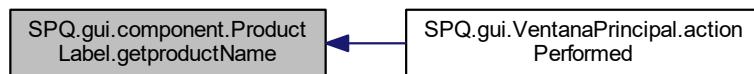
Definition at line 199 of file ProductLabel.java.

#### 7.37.3.5 getproductName()

```
JLabel SPQ.gui.component.ProductLabel.getproductName ( )
```

Definition at line 192 of file ProductLabel.java.

Here is the caller graph for this function:

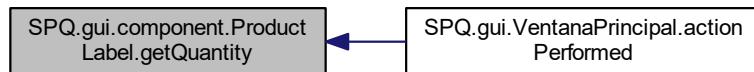


### 7.37.3.6 getQuantity()

```
JLabel SPQ.gui.component.ProductLabel.getQuantity ( )
```

Definition at line 207 of file ProductLabel.java.

Here is the caller graph for this function:

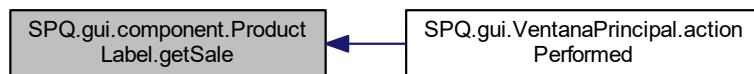


### 7.37.3.7 getSale()

```
JLabel SPQ.gui.component.ProductLabel.getSale ( )
```

Definition at line 184 of file ProductLabel.java.

Here is the caller graph for this function:



### 7.37.3.8 setbAdd()

```
void SPQ.gui.component.ProductLabel.setbAdd ( JButton bAdd )
```

Definition at line 219 of file ProductLabel.java.

#### 7.37.3.9 setbSub()

```
void SPQ.gui.component.ProductLabel.setbSub (
    JButton bSub )
```

Definition at line 227 of file ProductLabel.java.

#### 7.37.3.10 setPrice()

```
void SPQ.gui.component.ProductLabel.setPrice (
    JLabel price )
```

Definition at line 203 of file ProductLabel.java.

#### 7.37.3.11 setproductName()

```
void SPQ.gui.component.ProductLabel.setproductName (
    JLabel productName )
```

Definition at line 195 of file ProductLabel.java.

#### 7.37.3.12 setQuantity()

```
void SPQ.gui.component.ProductLabel.setQuantity (
    JLabel quantity )
```

Definition at line 211 of file ProductLabel.java.

#### 7.37.3.13 setSale()

```
void SPQ.gui.component.ProductLabel.setSale (
    JLabel sale )
```

Definition at line 188 of file ProductLabel.java.

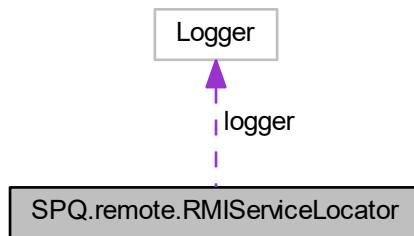
The documentation for this class was generated from the following file:

- NeverEmptyClient/src/main/java/SPQ/gui/component/[ProductLabel.java](#)

## 7.38 SPQ.remote.RMIServiceLocator Class Reference

[RMIServiceLocator](#) prepares the connection to the server side by founding the [NeverEmptyFacade](#) in the rmi registry.

Collaboration diagram for SPQ.remote.RMIServiceLocator:



### Public Member Functions

- [INeverEmptyFacade getNeverEmptyServer \(\)](#)
- [void setService \(String\[\] args\)](#)

#### 7.38.1 Detailed Description

[RMIServiceLocator](#) prepares the connection to the server side by founding the [NeverEmptyFacade](#) in the rmi registry.

Definition at line 13 of file `RMIServiceLocator.java`.

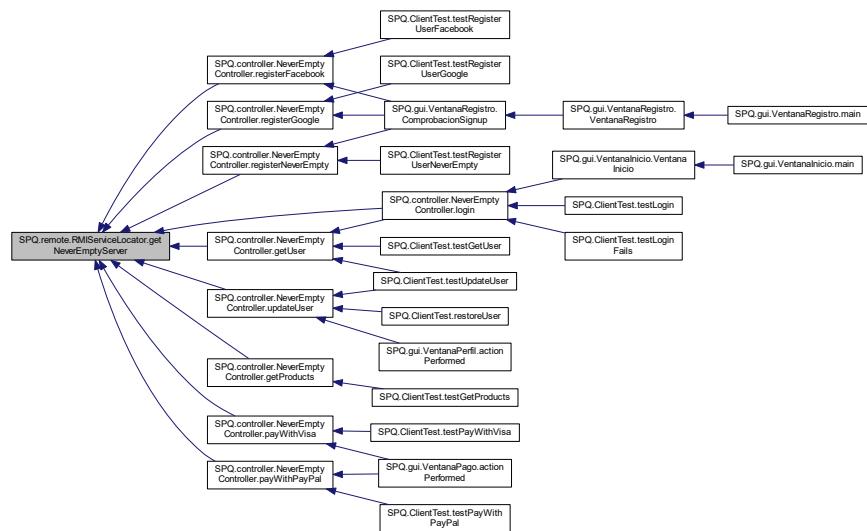
#### 7.38.2 Member Function Documentation

##### 7.38.2.1 `getNeverEmptyServer()`

```
INeverEmptyFacade SPQ.remote.RMIServiceLocator.getNeverEmptyServer ( )
```

Definition at line 19 of file `RMIServiceLocator.java`.

Here is the caller graph for this function:



### 7.38.2.2 setService()

```
void SPQ.remote.RMIServiceLocator.setService (
    String [] args )
```

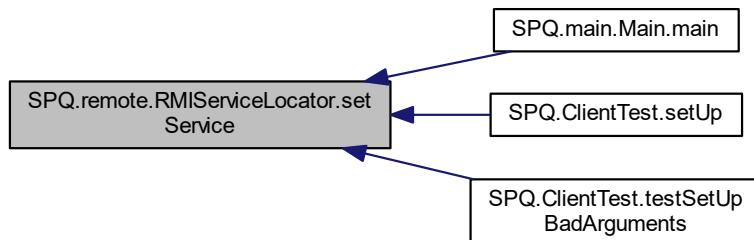
Prepares the connection to the server side by finding the [NeverEmptyFacade](#) in the rmi registry.

#### Parameters

<code>args</code>	<input type="text"/>
-------------------	----------------------

Definition at line 27 of file `RMIServiceLocator.java`.

Here is the caller graph for this function:

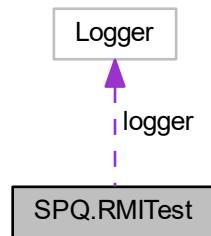


The documentation for this class was generated from the following file:

- NeverEmptyClient/src/main/java/SPQ/remote/RMIServiceLocator.java

## 7.39 SPQ.RMITest Class Reference

Collaboration diagram for SPQ.RMITest:



### Public Member Functions

- void `setUpClient()`
- void `loginNewUserTest()`
- void `loginExistingUserTest()`
- void `registerGoogleExistingUserTest()`
- void `registerFacebookExistingUserTest()`
- void `registerUserGoogleTest()`
- void `registerUserFacebookTest()`
- void `testPayWithPayPal()`
- void `testPayWithVisa()`
- void `testGetProducts()`
- void `deleteDatabase()`

### Static Public Member Functions

- static junit.framework.Test `suite()`
- static void `setUp()`
- static void `tearDown()`

#### 7.39.1 Detailed Description

Definition at line 34 of file `RMITest.java`.

## 7.39.2 Member Function Documentation

### 7.39.2.1 deleteDatabase()

```
void SPQ.RMITest.deleteDatabase ( )
```

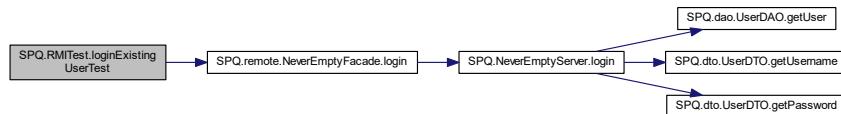
Definition at line 300 of file RMITest.java.

### 7.39.2.2 loginExistingUserTest()

```
void SPQ.RMITest.loginExistingUserTest ( )
```

Definition at line 167 of file RMITest.java.

Here is the call graph for this function:

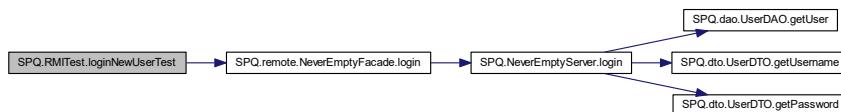


### 7.39.2.3 loginNewUserTest()

```
void SPQ.RMITest.loginNewUserTest ( )
```

Definition at line 141 of file RMITest.java.

Here is the call graph for this function:

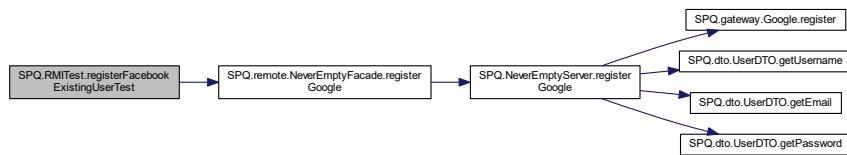


#### 7.39.2.4 registerFacebookExistingUserTest()

```
void SPQ.RMITest.registerFacebookExistingUserTest( )
```

Definition at line 208 of file RMITest.java.

Here is the call graph for this function:

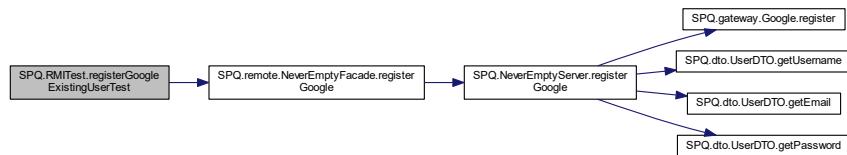


#### 7.39.2.5 registerGoogleExistingUserTest()

```
void SPQ.RMITest.registerGoogleExistingUserTest( )
```

Definition at line 187 of file RMITest.java.

Here is the call graph for this function:

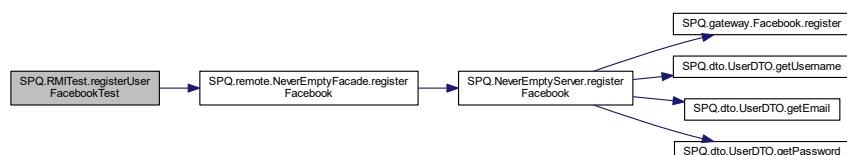


#### 7.39.2.6 registerUserFacebookTest()

```
void SPQ.RMITest.registerUserFacebookTest( )
```

Definition at line 250 of file RMITest.java.

Here is the call graph for this function:

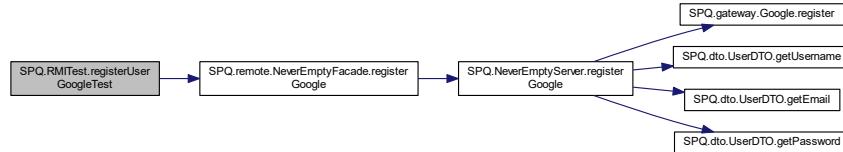


#### 7.39.2.7 registerUserGoogleTest()

```
void SPQ.RMITest.registerUserGoogleTest( )
```

Definition at line 230 of file RMITest.java.

Here is the call graph for this function:



#### 7.39.2.8 setUp()

```
static void SPQ.RMITest.setUp( ) [static]
```

Definition at line 48 of file RMITest.java.

#### 7.39.2.9 setUpClient()

```
void SPQ.RMITest.setUpClient( )
```

Definition at line 116 of file RMITest.java.

#### 7.39.2.10 suite()

```
static junit.framework.Test SPQ.RMITest.suite( ) [static]
```

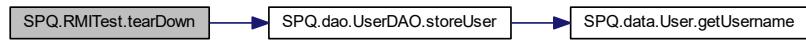
Definition at line 44 of file RMITest.java.

### 7.39.2.11 tearDown()

```
static void SPQ.RMITest.tearDown ( ) [static]
```

Definition at line 326 of file RMITest.java.

Here is the call graph for this function:



### 7.39.2.12 testGetProducts()

```
void SPQ.RMITest.testGetProducts ( )
```

Definition at line 289 of file RMITest.java.

Here is the call graph for this function:

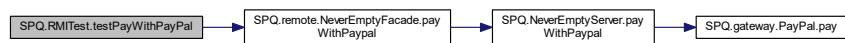


### 7.39.2.13 testPayWithPayPal()

```
void SPQ.RMITest.testPayWithPayPal ( )
```

Definition at line 267 of file RMITest.java.

Here is the call graph for this function:

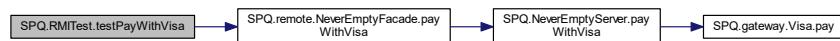


### 7.39.2.14 testPayWithVisa()

```
void SPQ.RMITest.testPayWithVisa( )
```

Definition at line 278 of file RMITest.java.

Here is the call graph for this function:



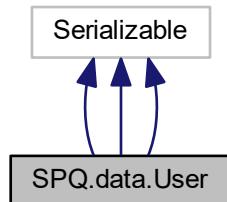
The documentation for this class was generated from the following file:

- NeverEmptyServer/src/test/java/SPQ/RMITest.java

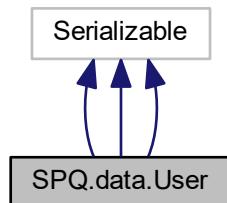
## 7.40 SPQ.data.User Class Reference

This class defines a [User](#).

Inheritance diagram for SPQ.data.User:



Collaboration diagram for SPQ.data.User:



## Public Member Functions

- `User (String email, String password)`
- `String getEmail ()`
- `void setEmail (String email)`
- `String getPassword ()`
- `void setPassword (String password)`
- `User (String email, String password)`
- `String getEmail ()`
- `void setEmail (String email)`
- `String getPassword ()`
- `void setPassword (String password)`
- `User (String username, String email, String password, String registerMethod)`
- `User (UserDTO userDTO)`
- `void updateUser (User user)`
- `String getAddress ()`
- `void setAddress (String address)`
- `String getCardholder ()`
- `void setCardholder (String cardholder)`
- `String getUsername ()`
- `void setUsername (String username)`
- `String getEmail ()`
- `void setEmail (String email)`
- `String getPassword ()`
- `void setPassword (String password)`
- `String getRegisterMethod ()`
- `void setRegisterMethod (String registerMethod)`
- `String getPayPalEmail ()`
- `void setPayPalEmail (String payPalEmail)`
- `String getPayPalPassword ()`
- `void setPayPalPassword (String payPalPassword)`
- `long getCardNumber ()`
- `void setCardNumber (long cardNumber)`
- `User (String email, String password, double balance)`
- `String getEmail ()`
- `void setEmail (String email)`
- `double getBalance ()`
- `void setBalance (double balance)`
- `String getPassword ()`
- `void setPassword (String password)`

### 7.40.1 Detailed Description

This class defines a `User`.

Definition at line 11 of file User.java.

### 7.40.2 Constructor & Destructor Documentation

**7.40.2.1 User() [1/5]**

```
SPQ.data.User.User (
    String email,
    String password )
```

Definition at line 16 of file User.java.

**7.40.2.2 User() [2/5]**

```
SPQ.data.User.User (
    String email,
    String password )
```

Definition at line 15 of file User.java.

**7.40.2.3 User() [3/5]**

```
SPQ.data.User.User (
    String username,
    String email,
    String password,
    String registerMethod )
```

Constructor of User.class

**Parameters**

<i>username</i>	
<i>email</i>	
<i>password</i>	
<i>registerMethod</i>	

Definition at line 44 of file User.java.

**7.40.2.4 User() [4/5]**

```
SPQ.data.User.User (
    UserDTO userDTO )
```

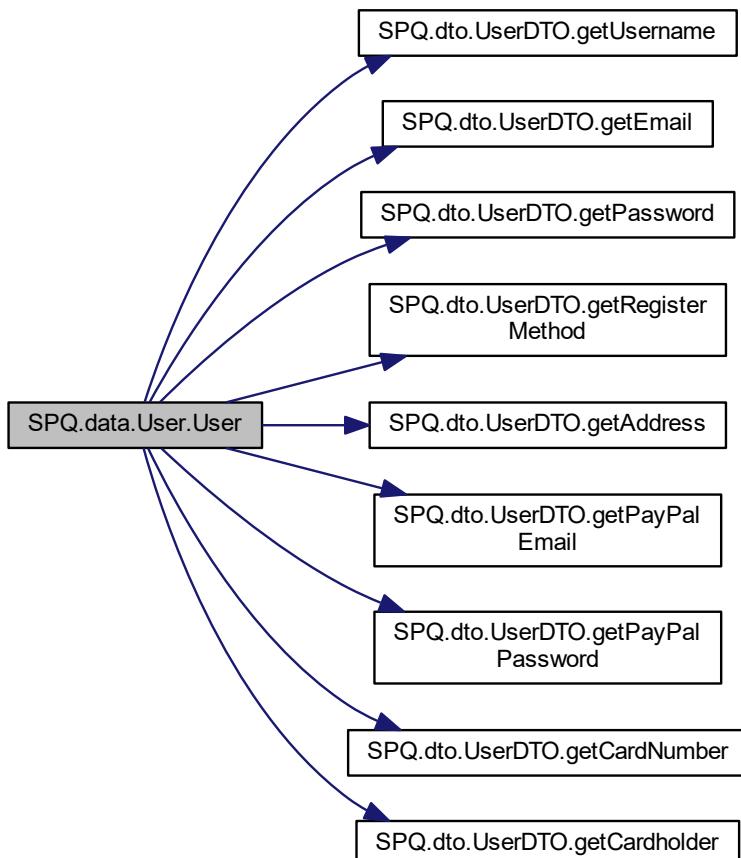
This method override the attributes of UserDTO

**Parameters**

<i>userDTO</i>	<input type="text"/>
----------------	----------------------

Definition at line 57 of file User.java.

Here is the call graph for this function:



#### 7.40.2.5 User() [5/5]

```
SPQ.data.User.User (
    String email,
    String password,
    double balance )
```

Definition at line 16 of file User.java.

### 7.40.3 Member Function Documentation

#### 7.40.3.1 getAddress()

```
String SPQ.data.User.getAddress ( )
```

Return the address attribute of the class

##### Returns

name instance

Definition at line 93 of file User.java.

Here is the caller graph for this function:



#### 7.40.3.2 getBalance()

```
double SPQ.data.User.getBalance ( )
```

Definition at line 29 of file User.java.

Here is the caller graph for this function:



### 7.40.3.3 getCardholder()

```
String SPQ.data.User.getCardholder ( )
```

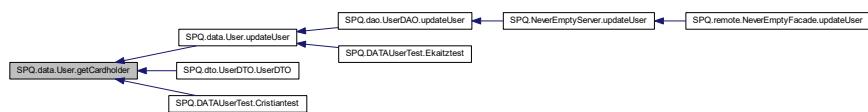
Return the cardholder attribute of the class

#### Returns

cardholder instance

Definition at line 108 of file User.java.

Here is the caller graph for this function:



### 7.40.3.4 getCardNumber()

```
long SPQ.data.User.getCardNumber ( )
```

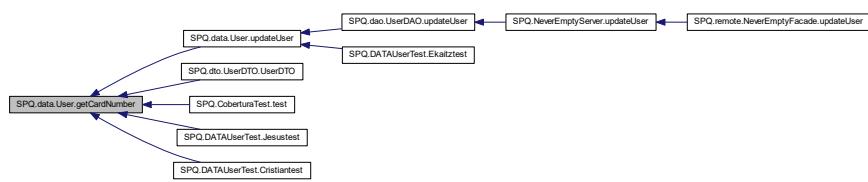
Return the cardnumber attribute of the class

#### Returns

cardnumber instance

Definition at line 213 of file User.java.

Here is the caller graph for this function:



#### 7.40.3.5 getEmail() [1/4]

```
String SPQ.data.User.getEmail ( )
```

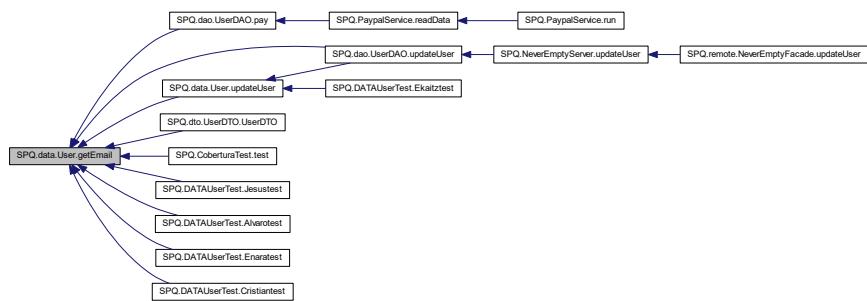
Definition at line 21 of file User.java.

#### 7.40.3.6 getEmail() [2/4]

```
String SPQ.data.User.getEmail ( )
```

Definition at line 22 of file User.java.

Here is the caller graph for this function:



#### 7.40.3.7 getEmail() [3/4]

```
String SPQ.data.User.getEmail ( )
```

Definition at line 23 of file User.java.

#### 7.40.3.8 getEmail() [4/4]

```
String SPQ.data.User.getEmail ( )
```

Return the email attribute of the class

##### Returns

email instance

Definition at line 138 of file User.java.

## 7.40.3.9 getPassword() [1/4]

```
String SPQ.data.User.getPassword ( )
```

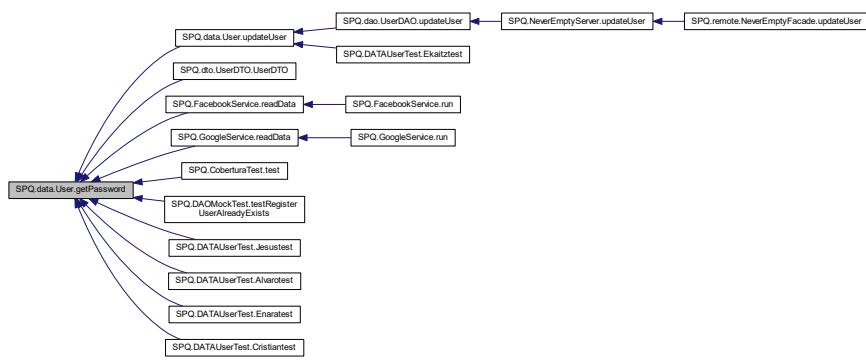
Definition at line 27 of file User.java.

## 7.40.3.10 getPassword() [2/4]

```
String SPQ.data.User.getPassword ( )
```

Definition at line 28 of file User.java.

Here is the caller graph for this function:



## 7.40.3.11 getPassword() [3/4]

```
String SPQ.data.User.getPassword ( )
```

Definition at line 35 of file User.java.

## 7.40.3.12 getPassword() [4/4]

```
String SPQ.data.User.getPassword ( )
```

Return the password attribute of the class

**Returns**

password instance

Definition at line 153 of file User.java.

#### 7.40.3.13 getPayPalEmail()

```
String SPQ.data.User.getPayPalEmail ( )
```

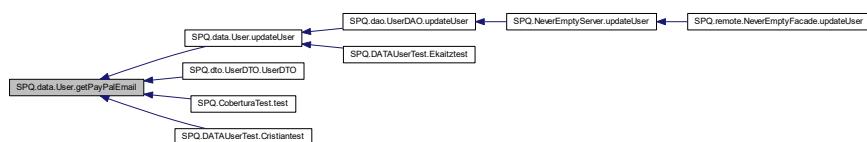
Return the paypalemail attribute of the class

##### Returns

paypalemail instance

Definition at line 183 of file User.java.

Here is the caller graph for this function:



#### 7.40.3.14 getPayPalPassword()

```
String SPQ.data.User.getPayPalPassword ( )
```

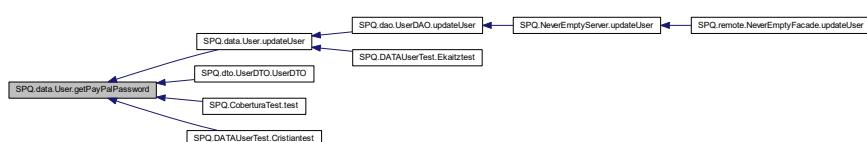
Return the paypalpassword attribute of the class

##### Returns

paypalpassword instance

Definition at line 198 of file User.java.

Here is the caller graph for this function:



#### 7.40.3.15 getRegisterMethod()

```
String SPQ.data.User.getRegisterMethod ( )
```

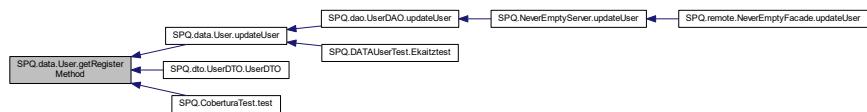
Return the registermethod attribute of the class

##### Returns

registermethod instance

Definition at line 168 of file User.java.

Here is the caller graph for this function:



#### 7.40.3.16 getUsername()

```
String SPQ.data.User.getUsername ( )
```

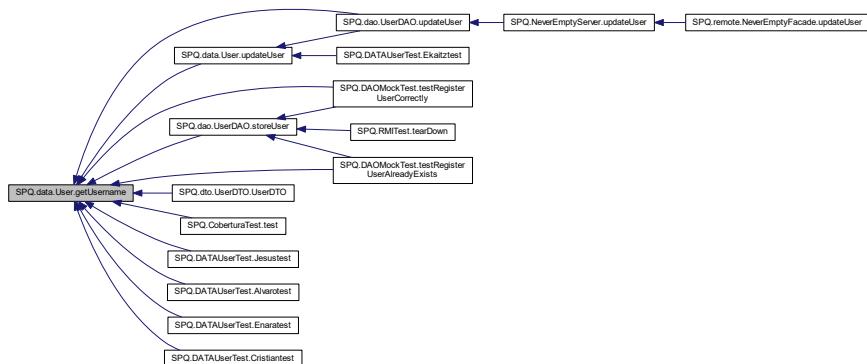
Return the username attribute of the class

##### Returns

username instance

Definition at line 123 of file User.java.

Here is the caller graph for this function:



#### 7.40.3.17 setAddress()

```
void SPQ.data.User setAddress (
    String address )
```

Sets address attribute of the class.

Definition at line 100 of file User.java.

Here is the caller graph for this function:



#### 7.40.3.18 setBalance()

```
void SPQ.data.User.setBalance (
    double balance )
```

Definition at line 32 of file User.java.

Here is the caller graph for this function:



#### 7.40.3.19 setCardholder()

```
void SPQ.data.User.setCardholder (
    String cardholder )
```

Sets cardholder attribute of the class.

Definition at line 115 of file User.java.

Here is the caller graph for this function:



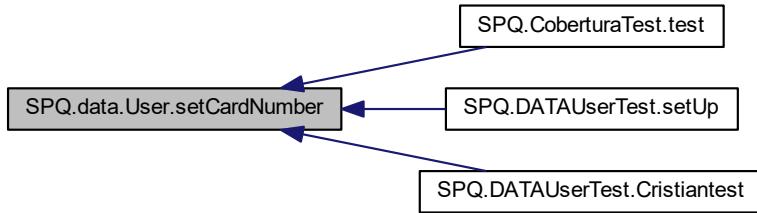
#### 7.40.3.20 setCardNumber()

```
void SPQ.data.User.setCardNumber (
    long cardNumber )
```

Sets cardnumber attribute of the class.

Definition at line 220 of file User.java.

Here is the caller graph for this function:



#### 7.40.3.21 setEmail() [1/4]

```
void SPQ.data.User.setEmail (
    String email )
```

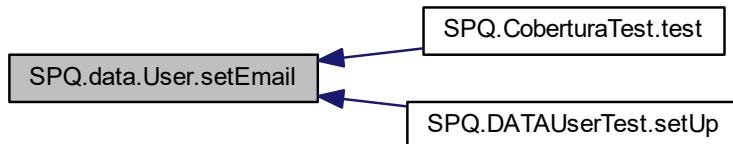
Definition at line 24 of file User.java.

#### 7.40.3.22 setEmail() [2/4]

```
void SPQ.data.User.setEmail (
    String email )
```

Definition at line 25 of file User.java.

Here is the caller graph for this function:



**7.40.3.23 setEmail() [3/4]**

```
void SPQ.data.User.setEmail (
    String email )
```

Definition at line 26 of file User.java.

**7.40.3.24 setEmail() [4/4]**

```
void SPQ.data.User.setEmail (
    String email )
```

Sets email attribute of the class.

Definition at line 145 of file User.java.

**7.40.3.25 setPassword() [1/4]**

```
void SPQ.data.User.setPassword (
    String password )
```

Definition at line 30 of file User.java.

**7.40.3.26 setPassword() [2/4]**

```
void SPQ.data.User.setPassword (
    String password )
```

Definition at line 31 of file User.java.

Here is the caller graph for this function:



**7.40.3.27 setPassword()** [3/4]

```
void SPQ.data.User.setPassword (
    String password )
```

Definition at line 38 of file User.java.

**7.40.3.28 setPassword()** [4/4]

```
void SPQ.data.User.setPassword (
    String password )
```

Sets password attribute of the class.

Definition at line 160 of file User.java.

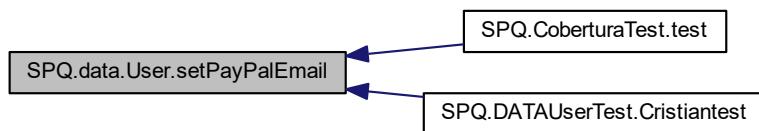
**7.40.3.29 setPayPalEmail()**

```
void SPQ.data.User.setPayPalEmail (
    String payPalEmail )
```

Sets paypalemail attribute of the class.

Definition at line 190 of file User.java.

Here is the caller graph for this function:



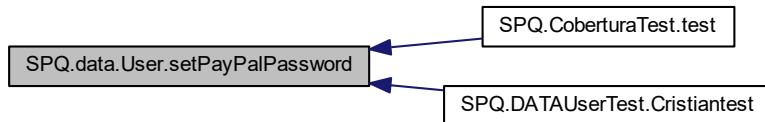
#### 7.40.3.30 setPayPalPassword()

```
void SPQ.data.User.setPayPalPassword (
    String paypalPassword )
```

Sets paypalpassword attribute of the class.

Definition at line 205 of file User.java.

Here is the caller graph for this function:



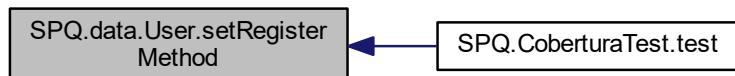
#### 7.40.3.31 setRegisterMethod()

```
void SPQ.data.User.setRegisterMethod (
    String registerMethod )
```

Sets registermethod attribute of the class.

Definition at line 175 of file User.java.

Here is the caller graph for this function:



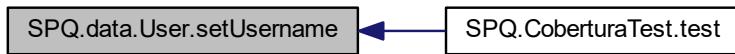
#### 7.40.3.32 setUsername()

```
void SPQ.data.User.setUsername (
    String username )
```

Sets username attribute of the class.

Definition at line 130 of file User.java.

Here is the caller graph for this function:

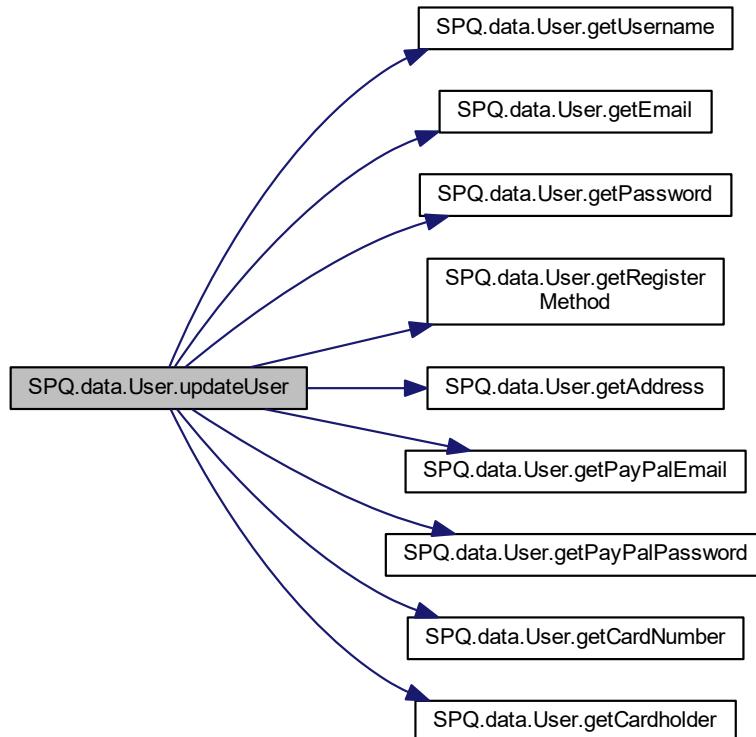


#### 7.40.3.33 updateUser()

```
void SPQ.data.User.updateUser (
    User user )
```

Definition at line 74 of file User.java.

Here is the call graph for this function:



Here is the caller graph for this function:

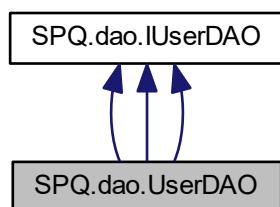


The documentation for this class was generated from the following file:

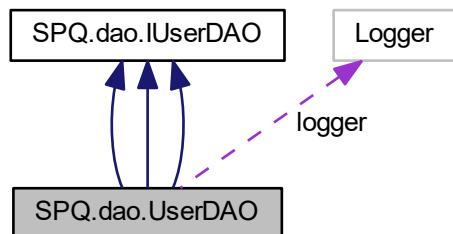
- FacebookService/src/main/java/SPQ/data/User.java

## 7.41 SPQ.dao.UserDAO Class Reference

Inheritance diagram for SPQ.dao.UserDAO:



Collaboration diagram for SPQ.dao.UserDAO:



## Public Member Functions

- [UserDAO \(\)](#)
- [User getUser \(String email\)](#)
- [UserDAO \(\)](#)
- [User getUser \(String email\)](#)
- [UserDAO \(\)](#)
- boolean [updateUser \(User user\)](#)
- boolean [storeUser \(User user\)](#)
- [User getUser \(String username, String password\)](#)
- [UserDAO \(\)](#)
- boolean [pay \(User user, double total\)](#)
- void [storeUser \(User user\)](#)

## Static Public Member Functions

- static void [main \(String\[ \] args\)](#)

### 7.41.1 Detailed Description

Definition at line 13 of file UserDAO.java.

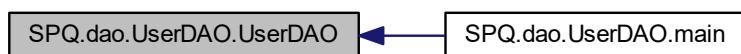
### 7.41.2 Constructor & Destructor Documentation

#### 7.41.2.1 UserDAO() [1/4]

SPQ.dao.UserDAO.UserDAO ( )

Definition at line 16 of file UserDAO.java.

Here is the caller graph for this function:



#### 7.41.2.2 `UserDAO()` [2/4]

```
SPQ.dao.UserDAO.UserDAO ( )
```

Definition at line 16 of file UserDAO.java.

#### 7.41.2.3 `UserDAO()` [3/4]

```
SPQ.dao.UserDAO.UserDAO ( )
```

Definition at line 20 of file UserDAO.java.

#### 7.41.2.4 `UserDAO()` [4/4]

```
SPQ.dao.UserDAO.UserDAO ( )
```

Definition at line 16 of file UserDAO.java.

### 7.41.3 Member Function Documentation

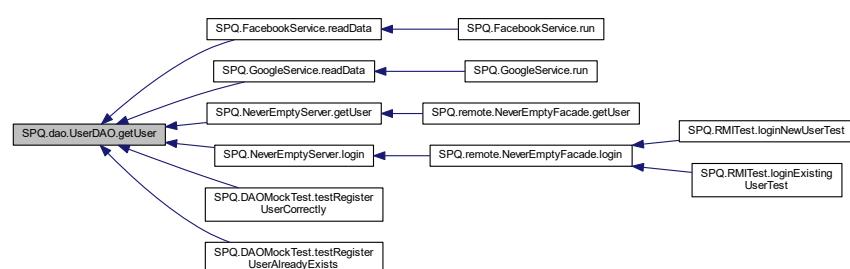
#### 7.41.3.1 `getUser()` [1/3]

```
User SPQ.dao.UserDAO.getUser (
    String email )
```

Implements [SPQ.dao.IUserDAO](#).

Definition at line 22 of file UserDAO.java.

Here is the caller graph for this function:



### 7.41.3.2 getUser() [2/3]

```
User SPQ.dao.UserDAO.getUser (
    String email )
```

Implements [SPQ.dao.IUserDAO](#).

Definition at line 22 of file UserDAO.java.

### 7.41.3.3 getUser() [3/3]

```
User SPQ.dao.UserDAO.getUser (
    String username,
    String password )
```

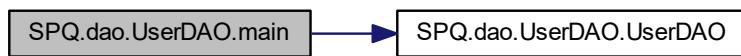
Definition at line 79 of file UserDAO.java.

### 7.41.3.4 main()

```
static void SPQ.dao.UserDAO.main (
    String [] args ) [static]
```

Definition at line 85 of file UserDAO.java.

Here is the call graph for this function:



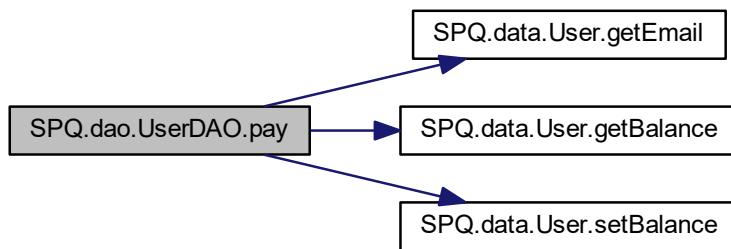
#### 7.41.3.5 pay()

```
boolean SPQ.dao.UserDAO.pay (
    User user,
    double total )
```

Implements [SPQ.dao.IUserDAO](#).

Definition at line 21 of file `UserDAO.java`.

Here is the call graph for this function:



Here is the caller graph for this function:



#### 7.41.3.6 storeUser() [1/2]

```
boolean SPQ.dao.UserDAO.storeUser (
    User user )
```

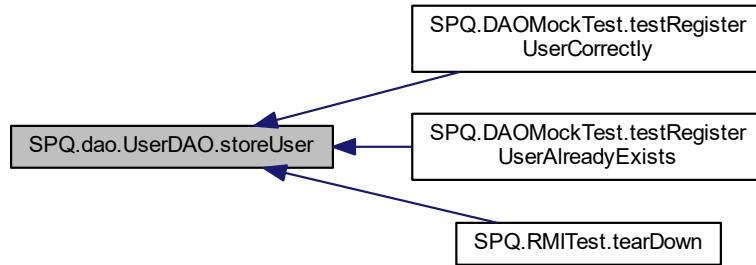
Implements [SPQ.dao.IUserDAO](#).

Definition at line 55 of file `UserDAO.java`.

Here is the call graph for this function:



Here is the caller graph for this function:



#### 7.41.3.7 `storeUser()` [2/2]

```
void SPQ.dao.UserDAO.storeUser (
    User user )
```

Implements `SPQ.dao.IUserDAO`.

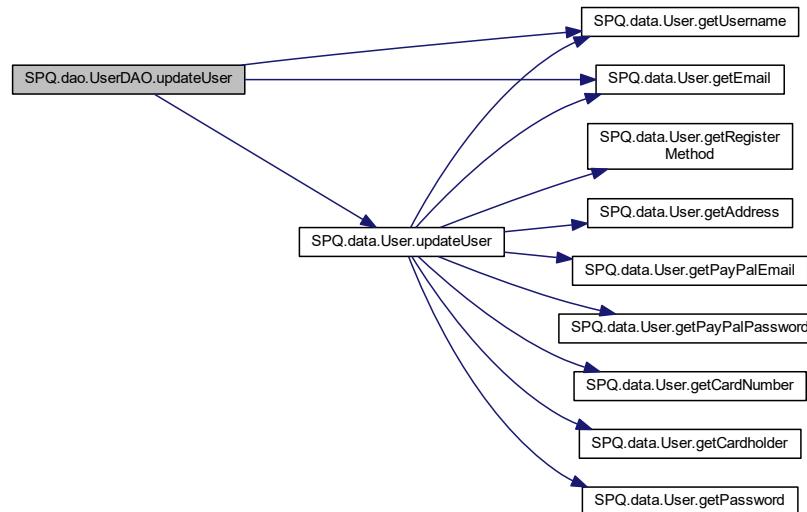
Definition at line 65 of file `UserDAO.java`.

#### 7.41.3.8 `updateUser()`

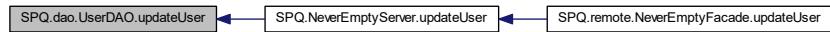
```
boolean SPQ.dao.UserDAO.updateUser (
    User user )
```

Definition at line 24 of file `UserDAO.java`.

Here is the call graph for this function:



Here is the caller graph for this function:



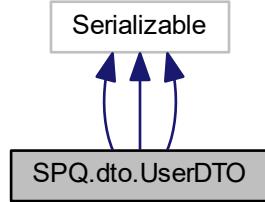
The documentation for this class was generated from the following file:

- FacebookService/src/main/java/SPQ/dao/[UserDAO.java](#)

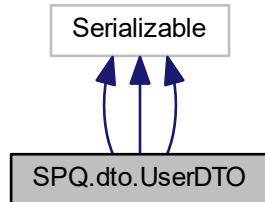
## 7.42 SPQ.dto.UserDTO Class Reference

This class prepares a user to be transferred.

Inheritance diagram for SPQ.dto.UserDTO:



Collaboration diagram for SPQ.dto.UserDTO:



## Public Member Functions

- `UserDTO (User user)`
- `UserDTO (String password, String email)`
- `UserDTO (String email, String password, String registerMethod)`
- `String getEmail ()`
- `void setEmail (String email)`
- `String getPassword ()`
- `void setPassword (String password)`
- `UserDTO (User user)`
- `UserDTO (String password, String email)`
- `UserDTO (String email, String password, String registerMethod)`
- `String getEmail ()`
- `void setEmail (String email)`
- `String getPassword ()`
- `void setPassword (String password)`
- `UserDTO (User user)`
- `String getAddress ()`
- `void setAddress (String address)`
- `String getCardholder ()`
- `void setCardholder (String cardholder)`
- `UserDTO (String username, String password)`
- `UserDTO (String username, String password, String email)`
- `UserDTO (String username, String email, String password, String registerMethod, String address, String payPalEmail, String payPalPassword, long cardNumber, String cardholder)`
- `UserDTO (String username, String email, String password, String registerMethod)`
- `String getUsername ()`
- `void setUsername (String username)`
- `String getEmail ()`
- `void setEmail (String email)`
- `String getPassword ()`
- `void setPassword (String password)`
- `String getRegisterMethod ()`
- `void setRegisterMethod (String registerMethod)`
- `String getPayPalEmail ()`
- `void setPayPalEmail (String payPalEmail)`
- `String getPayPalPassword ()`
- `void setPayPalPassword (String payPalPassword)`
- `long getCardNumber ()`
- `void setCardNumber (long cardNumber)`

### 7.42.1 Detailed Description

This class prepares a user to be transferred.

Definition at line 7 of file UserDTO.java.

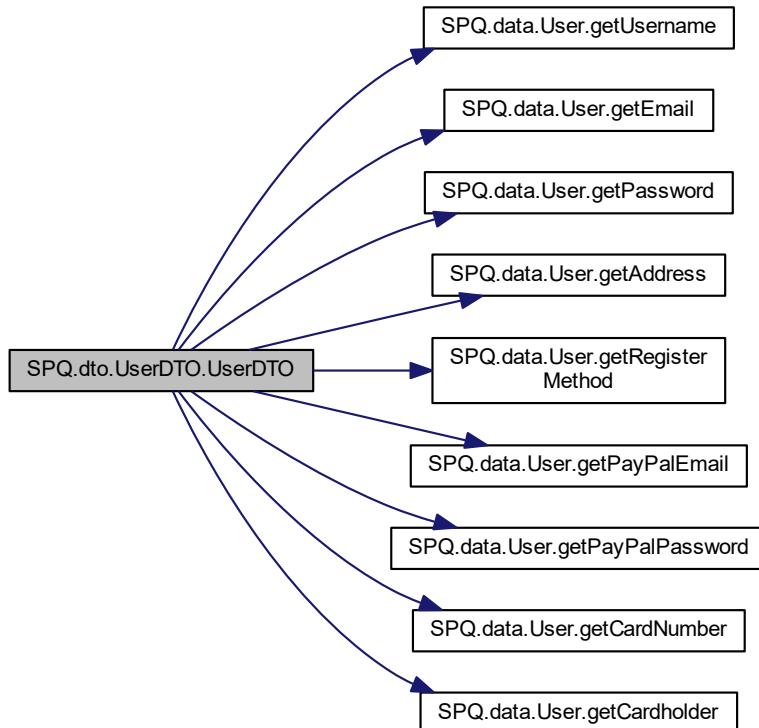
### 7.42.2 Constructor & Destructor Documentation

#### 7.42.2.1 UserDTO() [1/11]

```
SPQ.dto.UserDTO.UserDTO (
    User user )
```

Definition at line 14 of file UserDTO.java.

Here is the call graph for this function:



#### 7.42.2.2 UserDTO() [2/11]

```
SPQ.dto.UserDTO.UserDTO (
    String password,
    String email )
```

Definition at line 21 of file UserDTO.java.

**7.42.2.3 UserDTO() [3/11]**

```
SPQ.dto.UserDTO.UserDTO (
    String email,
    String password,
    String registerMethod )
```

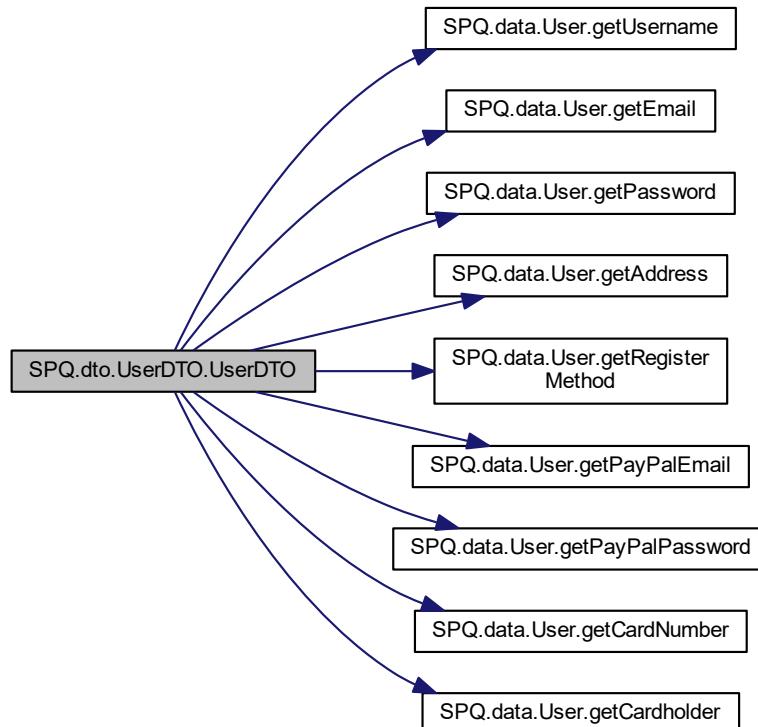
Definition at line 27 of file UserDTO.java.

**7.42.2.4 UserDTO() [4/11]**

```
SPQ.dto.UserDTO.UserDTO (
    User user )
```

Definition at line 14 of file UserDTO.java.

Here is the call graph for this function:



**7.42.2.5 UserDTO() [5/11]**

```
SPQ.dto.UserDTO.UserDTO (
    String password,
    String email )
```

Definition at line 21 of file UserDTO.java.

**7.42.2.6 UserDTO() [6/11]**

```
SPQ.dto.UserDTO.UserDTO (
    String email,
    String password,
    String registerMethod )
```

Definition at line 27 of file UserDTO.java.

**7.42.2.7 UserDTO() [7/11]**

```
SPQ.dto.UserDTO.UserDTO (
    User user )
```

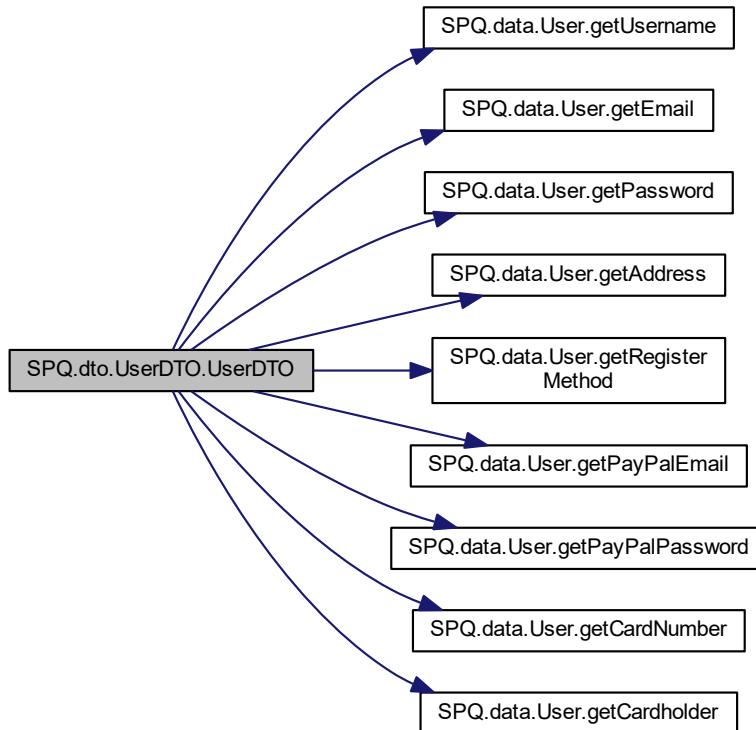
Constructor of [UserDTO](#)

**Parameters**

<i>user</i>	<input type="button" value=""/>
-------------	---------------------------------

Definition at line 35 of file UserDTO.java.

Here is the call graph for this function:



#### 7.42.2.8 UserDTO() [8/11]

```
SPQ.dto.UserDTO.UserDTO (
    String username,
    String password )
```

Constructor of [UserDTO](#)

Parameters

<code>username</code>	
<code>password</code>	

Definition at line 85 of file `UserDTO.java`.

#### 7.42.2.9 UserDTO() [9/11]

```
SPQ.dto.UserDTO.UserDTO (
    String username,
```

```
        String password,
        String email )
```

Constructor of [UserDTO](#)

#### Parameters

<i>username</i>	
<i>password</i>	
<i>email</i>	

Definition at line 97 of file UserDTO.java.

#### 7.42.2.10 UserDTO() [10/11]

```
SPQ.dto.UserDTO.UserDTO (
    String username,
    String email,
    String password,
    String registerMethod,
    String address,
    String payPalEmail,
    String payPalPassword,
    long cardNumber,
    String cardholder )
```

Constructor of [UserDTO](#)

#### Parameters

<i>username</i>	
<i>email</i>	
<i>password</i>	
<i>registerMethod</i>	
<i>address</i>	
<i>payPalEmail</i>	
<i>payPalPassword</i>	
<i>cardNumber</i>	
<i>cardholder</i>	

Definition at line 116 of file UserDTO.java.

#### 7.42.2.11 UserDTO() [11/11]

```
SPQ.dto.UserDTO.UserDTO (
    String username,
    String email,
```

```
String password,  
String registerMethod )
```

Constructor of UserDTO

#### Parameters

<i>username</i>	
<i>email</i>	
<i>password</i>	
<i>registerMethod</i>	

Definition at line 137 of file UserDTO.java.

### 7.42.3 Member Function Documentation

#### 7.42.3.1 getAddress()

```
String SPQ.dto.UserDTO.getAddress ( )
```

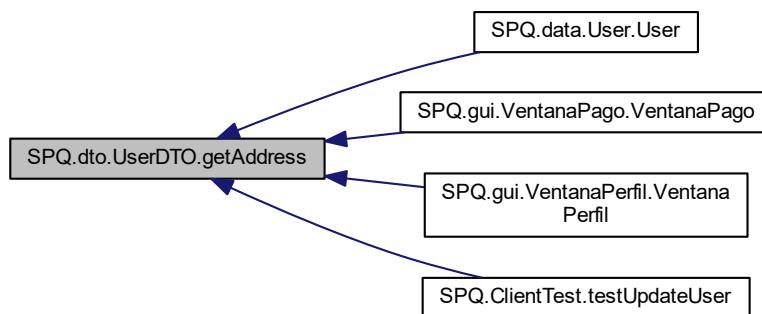
Returns the address attribute of the class.

#### Returns

address instance.

Definition at line 52 of file UserDTO.java.

Here is the caller graph for this function:



#### 7.42.3.2 getCardholder()

```
String SPQ.dto.UserDTO.getCardholder ( )
```

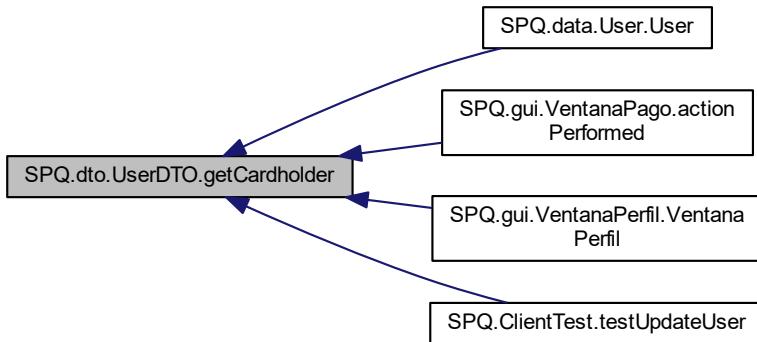
Returns the cardholder attribute of the class.

##### Returns

cardholder instance.

Definition at line 68 of file UserDTO.java.

Here is the caller graph for this function:



#### 7.42.3.3 getCardNumber()

```
long SPQ.dto.UserDTO.getCardNumber ( )
```

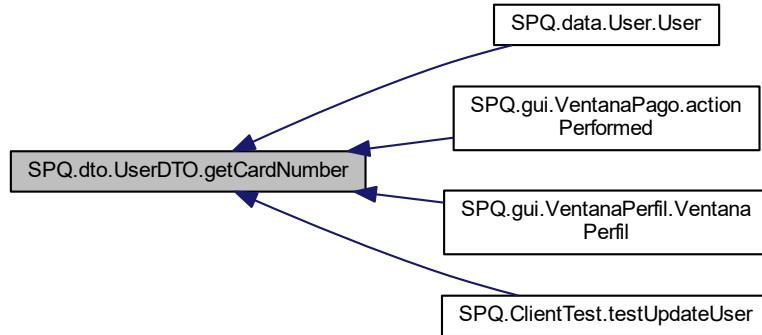
Returns the cardnumber attribute of the class.

##### Returns

cardnumber instance.

Definition at line 239 of file UserDTO.java.

Here is the caller graph for this function:



#### 7.42.3.4 getEmail() [1/3]

```
String SPQ.dto.UserDTO.getEmail ( )
```

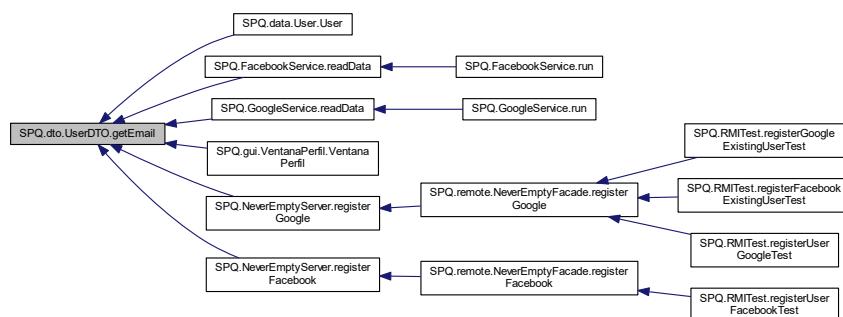
Definition at line 34 of file UserDTO.java.

#### 7.42.3.5 getEmail() [2/3]

```
String SPQ.dto.UserDTO.getEmail ( )
```

Definition at line 34 of file UserDTO.java.

Here is the caller graph for this function:



#### 7.42.3.6 getEmail() [3/3]

```
String SPQ.dto.UserDTO.getEmail ( )
```

Returns the email attribute of the class.

##### Returns

email instance.

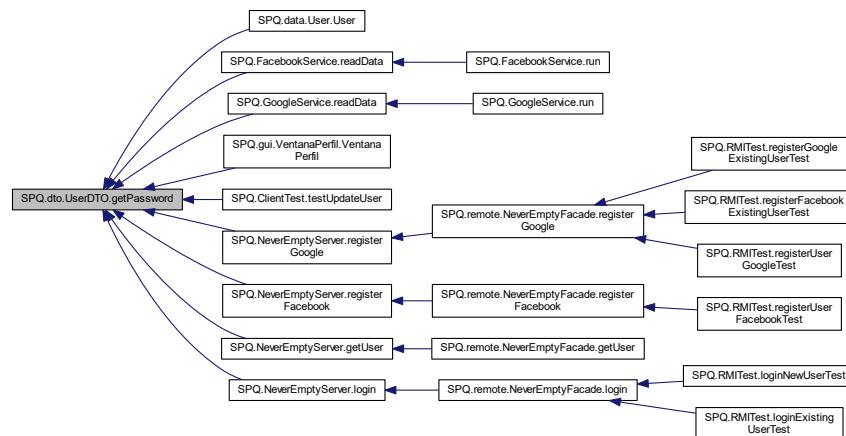
Definition at line 164 of file UserDTO.java.

#### 7.42.3.7 getPassword() [1/3]

```
String SPQ.dto.UserDTO.getPassword ( )
```

Definition at line 42 of file UserDTO.java.

Here is the caller graph for this function:



#### 7.42.3.8 getPassword() [2/3]

```
String SPQ.dto.UserDTO.getPassword ( )
```

Definition at line 42 of file UserDTO.java.

#### 7.42.3.9 getPassword() [3/3]

```
String SPQ.dto.UserDTO.getPassword( )
```

Returns the password attribute of the class.

##### Returns

password instance.

Definition at line 179 of file UserDTO.java.

#### 7.42.3.10 getPayPalEmail()

```
String SPQ.dto.UserDTO.getPayPalEmail( )
```

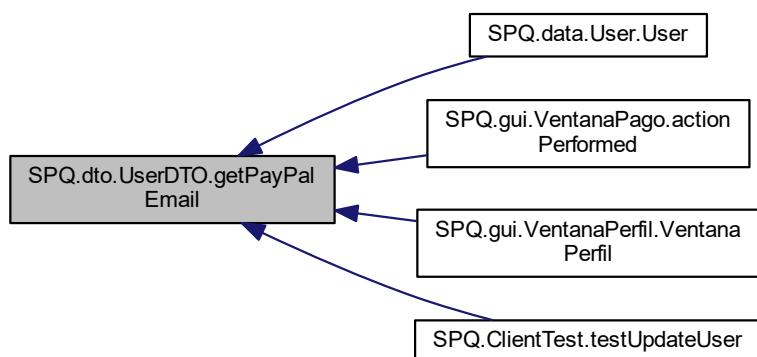
Returns the paypalEmail attribute of the class.

##### Returns

cardnumber instance.

Definition at line 209 of file UserDTO.java.

Here is the caller graph for this function:



#### 7.42.3.11 getPayPalPassword()

```
String SPQ.dto.UserDTO.getPayPalPassword ( )
```

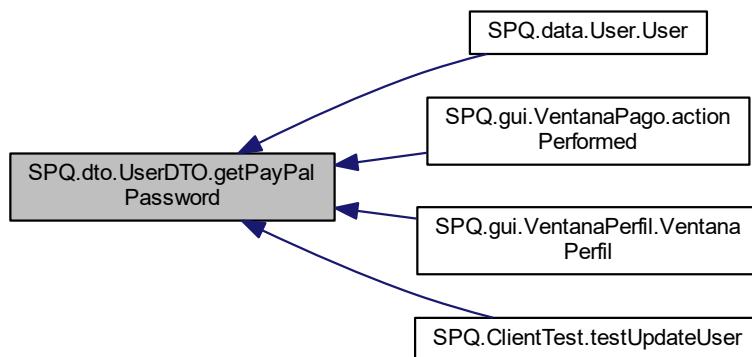
Returns the getPayPal attribute of the class.

##### Returns

GetPayPal instance.

Definition at line 224 of file UserDTO.java.

Here is the caller graph for this function:



#### 7.42.3.12 getRegisterMethod()

```
String SPQ.dto.UserDTO.getRegisterMethod ( )
```

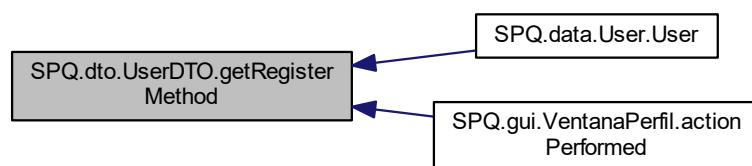
Returns the registerMethod attribute of the class.

##### Returns

registerMethod instance.

Definition at line 194 of file UserDTO.java.

Here is the caller graph for this function:



#### 7.42.3.13 getUsername()

```
String SPQ.dto.UserDTO.getUsername ( )
```

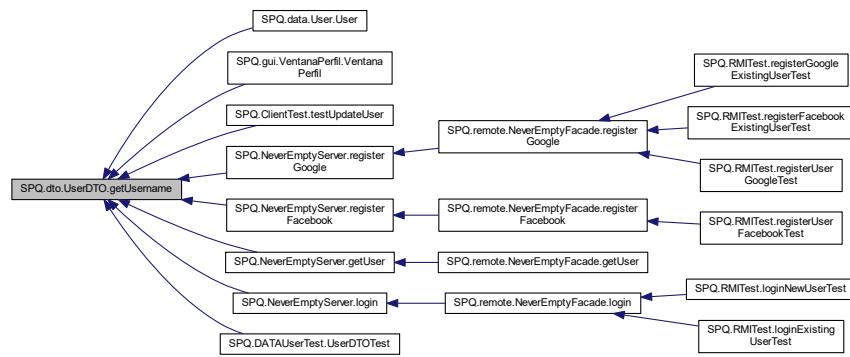
Returns the username attribute of the class.

##### Returns

username instance.

Definition at line 149 of file UserDTO.java.

Here is the caller graph for this function:



#### 7.42.3.14 setAddress()

```
void SPQ.dto.UserDTO.setAddress (
    String address )
```

Sets address attribute of the class.

Definition at line 59 of file UserDTO.java.

#### 7.42.3.15 setCardholder()

```
void SPQ.dto.UserDTO.setCardholder (
    String cardholder )
```

Sets cardholder attribute of the class.

Definition at line 75 of file UserDTO.java.

**7.42.3.16 setCardNumber()**

```
void SPQ.dto.UserDTO.setCardNumber (
    long cardNumber )
```

Sets cardNumber attribute of the class.

Definition at line 246 of file UserDTO.java.

**7.42.3.17 setEmail() [1/3]**

```
void SPQ.dto.UserDTO.setEmail (
    String email )
```

Definition at line 38 of file UserDTO.java.

**7.42.3.18 setEmail() [2/3]**

```
void SPQ.dto.UserDTO.setEmail (
    String email )
```

Definition at line 38 of file UserDTO.java.

**7.42.3.19 setEmail() [3/3]**

```
void SPQ.dto.UserDTO.setEmail (
    String email )
```

Sets email attribute of the class.

Definition at line 171 of file UserDTO.java.

**7.42.3.20 setPassword() [1/3]**

```
void SPQ.dto.UserDTO.setPassword (
    String password )
```

Definition at line 46 of file UserDTO.java.

**7.42.3.21 setPassword() [2/3]**

```
void SPQ.dto.UserDTO.setPassword (
    String password )
```

Definition at line 46 of file UserDTO.java.

**7.42.3.22 setPassword() [3/3]**

```
void SPQ.dto.UserDTO.setPassword (
    String password )
```

Sets password attribute of the class.

Definition at line 186 of file UserDTO.java.

**7.42.3.23 setPayPalEmail()**

```
void SPQ.dto.UserDTO.setPayPalEmail (
    String payPalEmail )
```

Sets paypalEmail attribute of the class.

Definition at line 216 of file UserDTO.java.

**7.42.3.24 setPayPalPassword()**

```
void SPQ.dto.UserDTO.setPayPalPassword (
    String payPalPassword )
```

Sets paypalPassword attribute of the class.

Definition at line 231 of file UserDTO.java.

**7.42.3.25 setRegisterMethod()**

```
void SPQ.dto.UserDTO.setRegisterMethod (
    String registerMethod )
```

Sets registerMethod attribute of the class.

Definition at line 201 of file UserDTO.java.

#### 7.42.3.26 setUsername()

```
void SPQ.dto.UserDTO.setUsername (
    String username )
```

Sets username attribute of the class.

Definition at line 156 of file UserDTO.java.

The documentation for this class was generated from the following file:

- FacebookService/src/main/java/SPQ/dto/[UserDTO.java](#)

## 7.43 SPQ.Utilities Class Reference

[Utilities](#) class. The [Utilities](#) class gathers reusable utility methods in different points of the project.

### Public Member Functions

- ImageIcon [getImageFromResources](#) (String filename)

#### 7.43.1 Detailed Description

[Utilities](#) class. The [Utilities](#) class gathers reusable utility methods in different points of the project.

Definition at line 15 of file Utilities.java.

#### 7.43.2 Member Function Documentation

##### 7.43.2.1 [getImageFromResources\(\)](#)

```
ImageIcon SPQ.Utilities.getImageFromResources (
    String filename )
```

Gets an image from /resource/images/ with the especified name.

#### Parameters

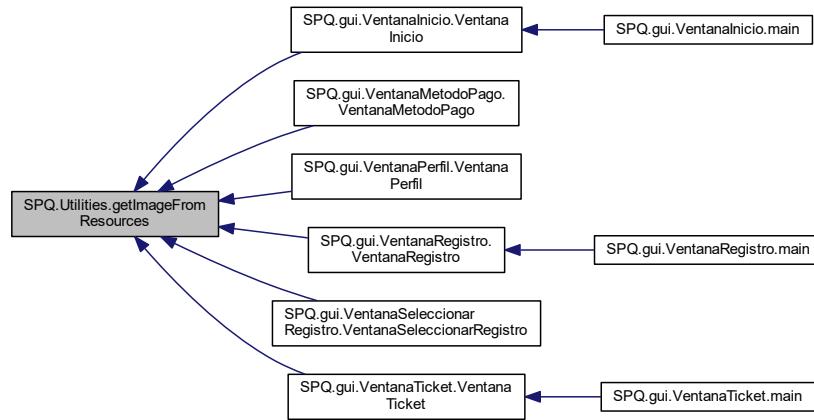
<i>filename</i>	. Image name and extension.
-----------------	-----------------------------

**Returns**

`ImageIcon`. Returns an image from `/resources/images/` as a `ImageIcon` or null if it can not be found.

Definition at line 21 of file `Utilities.java`.

Here is the caller graph for this function:



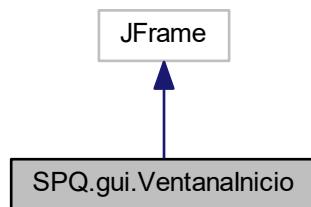
The documentation for this class was generated from the following file:

- `NeverEmptyClient/src/main/java/SPQ/Utilities.java`

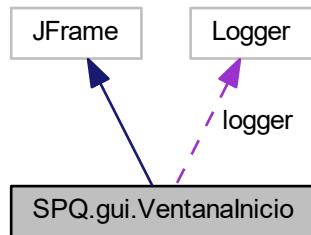
## 7.44 SPQ.gui.Ventanalinicio Class Reference

This is the first window of the application. It contains a form with username and password fields to login. And two buttons: Login and Sign Up. The login button (`bLogin`) calls the login function of the controller and sends the form data. The sign up (`bSignUp`) button calls "VentanaSeleccionarRegistro" window.

Inheritance diagram for `SPQ.gui.Ventanalinicio`:



Collaboration diagram for SPQ.gui.VentanaInicio:



## Public Member Functions

- `VentanaInicio (NeverEmptyController neverEmptyController)`
- `void iniciarAplicacion ()`

## Static Public Member Functions

- `static void main (String[] args)`

### 7.44.1 Detailed Description

This is the first window of the application. It contains a form with username and password fields to login. And two buttons: Login and Sign Up. The login button (bLogin) calls the login function of the controller and sends the form data. The sign up (bSignUp) button calls "VentanaSeleccionarRegistro" window.

Definition at line 35 of file VentanaInicio.java.

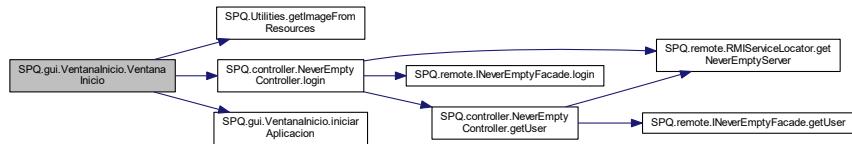
### 7.44.2 Constructor & Destructor Documentation

#### 7.44.2.1 VentanaInicio()

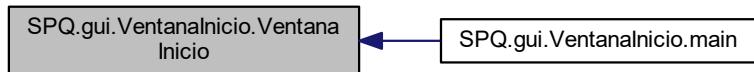
```
SPQ.gui.VentanaInicio.VentanaInicio (
    NeverEmptyController neverEmptyController )
```

Definition at line 54 of file VentanaInicio.java.

Here is the call graph for this function:



Here is the caller graph for this function:



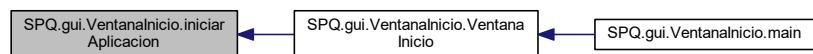
### 7.44.3 Member Function Documentation

#### 7.44.3.1 iniciarAplicacion()

```
void SPQ.gui.VentanaInicio.iniciarAplicacion ()
```

Definition at line 167 of file VentanaInicio.java.

Here is the caller graph for this function:

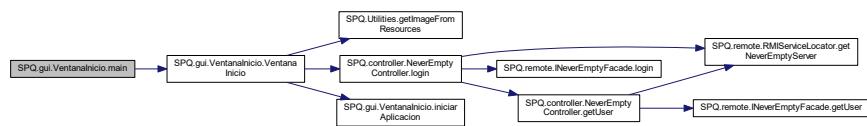


#### 7.44.3.2 main()

```
static void SPQ.gui.VentanaInicio.main (
    String [] args ) [static]
```

Definition at line 49 of file Ventanainicio.java.

Here is the call graph for this function:



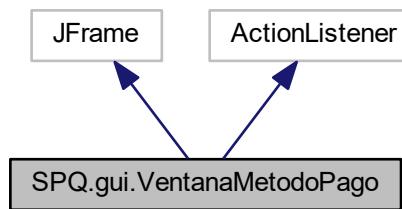
The documentation for this class was generated from the following file:

- NeverEmptyClient/src/main/java/SPQ/gui/[Ventanainicio.java](#)

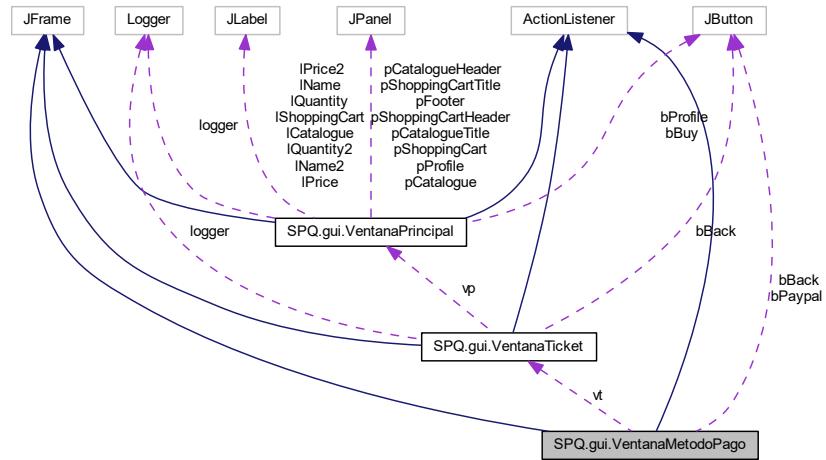
## 7.45 SPQ.gui.VentanaMetodoPago Class Reference

This window contains two main buttons, paypal (bPaypal) button and visa button (bVisa). Both of them call "← VentanaPago" but bPayPal passes "paypal" String to "VentanaPago" and bVisa "visa".

Inheritance diagram for SPQ.gui.VentanaMetodoPago:



## Collaboration diagram for SPQ.gui.VentanaMetodoPago:



## Public Member Functions

- `VentanaMetodoPago (NeverEmptyController neverEmptyController, double total, VentanaTicket ventanaTicket)`
  - void `actionPerformed (ActionEvent e)`

## Public Attributes

- VentanaTicket vt

#### **7.45.1 Detailed Description**

This window contains two main buttons, paypal (bPaypal) button and visa button (bVisa). Both of them call "VentanaPago" but bPayPal passes "paypal" String to "VentanaPago" and bVisa "visa".

Definition at line 25 of file VentanaMetodoPago.java.

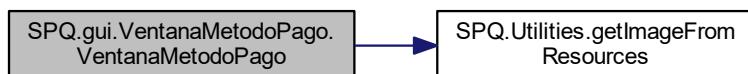
#### 7.45.2 Constructor & Destructor Documentation

#### 7.45.2.1 VentanaMetodoPago()

```
SPQ.gui.VentanaMetodoPago.VentanaMetodoPago (
    NeverEmptyController neverEmptyController,
    double total,
    VentanaTicket ventanaTicket )
```

Definition at line 35 of file VentanaMetodoPago.java.

Here is the call graph for this function:



#### 7.45.3 Member Function Documentation

##### 7.45.3.1 actionPerformed()

```
void SPQ.gui.VentanaMetodoPago.actionPerformed (
    ActionEvent e )
```

Definition at line 96 of file VentanaMetodoPago.java.

#### 7.45.4 Member Data Documentation

##### 7.45.4.1 vt

```
VentanaTicket SPQ.gui.VentanaMetodoPago.vt
```

Definition at line 32 of file VentanaMetodoPago.java.

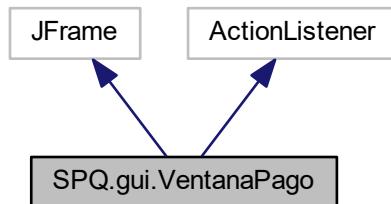
The documentation for this class was generated from the following file:

- NeverEmptyClient/src/main/java/SPQ/gui/[VentanaMetodoPago.java](#)

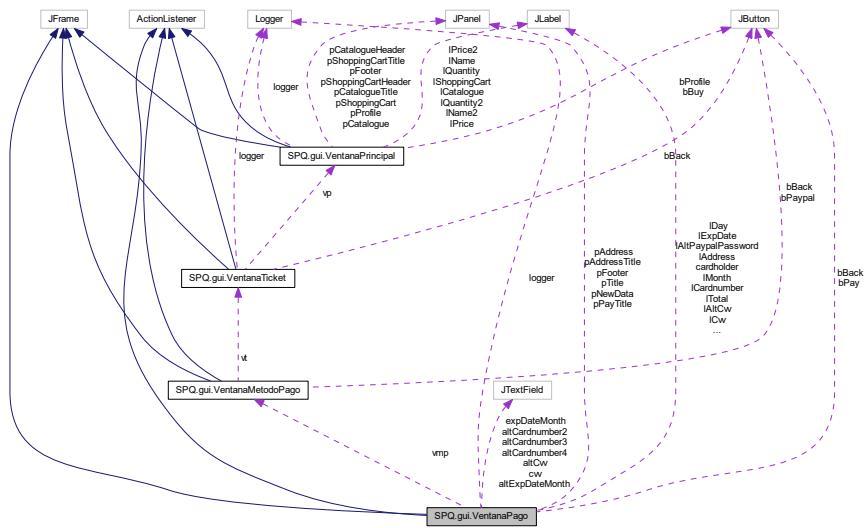
## 7.46 SPQ.gui.VentanaPago Class Reference

This window shows the last step to make a payment. [VentanaPago](#) has two modes: visa and paypal. Depending on the input attribute the form changes. On the top is always shown the shipping address that can be modified in "VentanaPerfil". The PayPal form consist of two fields: username and password. The VISA form consist of four fields: cardnumber, cardholder, expiration date and CVV. The left bottom button allows to use a new paypal user or visa data to pay. It calls "Animation" and transforms the form. The right bottom button calls the NeverEmptyController payWithPayPal(PaymentDTO) or payWithVisa(PaymentDTO) function.

## Inheritance diagram for SPQ.gui.VentanaPago:



## Collaboration diagram for SPQ.gui.VentanaPago:



## Classes

- class **Animate**

## Public Member Functions

- `VentanaPago (NeverEmptyController neverEmptyController, String paymentMethod, double total, VentanaMetodoPago ventanaMetodoPago)`
- `void actionPerformed (ActionEvent e)`

## Static Public Member Functions

- `static Rectangle calculateProgress (Rectangle startBounds, Rectangle targetBounds, double progress)`
- `static Point calculateProgress (Point startPoint, Point targetPoint, double progress)`
- `static int calculateProgress (int startValue, int endValue, double fraction)`
- `static Dimension calculateProgress (Dimension startSize, Dimension targetSize, double progress)`

## Static Public Attributes

- `static VentanaMetodoPago vmp`

### 7.46.1 Detailed Description

This window shows the last step to make a payment. `VentanaPago` has two modes: visa and paypal. Depending on the input attribute the form changes. On the top is always shown the shipping address that can be modified in "↔ VentanaPerfil". The PayPal form consist of two fields: username and password. The VISA form consist of four fields: cardnumber, cardholder, expiration date and CVV. The left bottom button allows to use a new paypal user or visa data to pay. It calls "Animation" and transforms the form. The right bottom button calls the NeverEmptyController `payWithPayPal(PaymentDTO)` or `payWithVisa(PaymentDTO)` function.

Definition at line 41 of file `VentanaPago.java`.

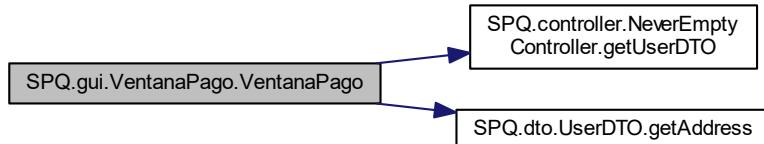
### 7.46.2 Constructor & Destructor Documentation

#### 7.46.2.1 `VentanaPago()`

```
SPQ.gui.VentanaPago.VentanaPago (
    NeverEmptyController neverEmptyController,
    String paymentMethod,
    double total,
    VentanaMetodoPago ventanaMetodoPago )
```

Definition at line 80 of file `VentanaPago.java`.

Here is the call graph for this function:



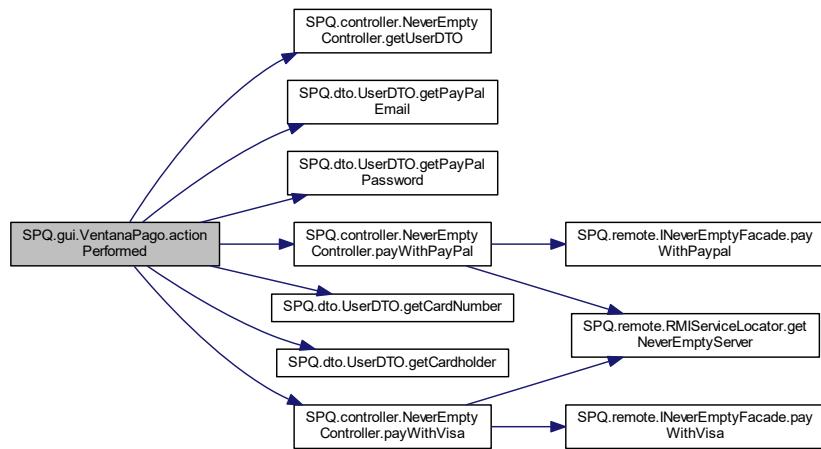
### 7.46.3 Member Function Documentation

#### 7.46.3.1 actionPerformed()

```
void SPQ.gui.VentanaPago.actionPerformed (
    ActionEvent e )
```

Definition at line 405 of file VentanaPago.java.

Here is the call graph for this function:

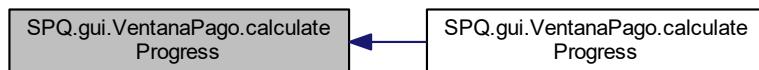


#### 7.46.3.2 calculateProgress() [1/4]

```
static Rectangle SPQ.gui.VentanaPago.calculateProgress (
    Rectangle startBounds,
    Rectangle targetBounds,
    double progress ) [static]
```

Definition at line 566 of file VentanaPago.java.

Here is the caller graph for this function:

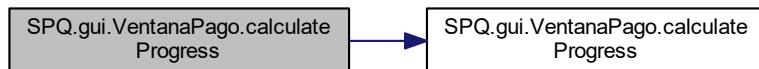


#### 7.46.3.3 calculateProgress() [2/4]

```
static Point SPQ.gui.VentanaPago.calculateProgress (
    Point startPoint,
    Point targetPoint,
    double progress ) [static]
```

Definition at line 581 of file VentanaPago.java.

Here is the call graph for this function:



#### 7.46.3.4 calculateProgress() [3/4]

```
static int SPQ.gui.VentanaPago.calculateProgress (
    int startValue,
    int endValue,
    double fraction ) [static]
```

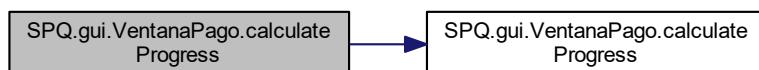
Definition at line 596 of file VentanaPago.java.

#### 7.46.3.5 calculateProgress() [4/4]

```
static Dimension SPQ.gui.VentanaPago.calculateProgress (
    Dimension startSize,
    Dimension targetSize,
    double progress ) [static]
```

Definition at line 607 of file VentanaPago.java.

Here is the call graph for this function:



#### 7.46.4 Member Data Documentation

##### 7.46.4.1 vmp

`VentanaMetodoPago` `SPQ.gui.VentanaPago.vmp` [static]

Definition at line 46 of file `VentanaPago.java`.

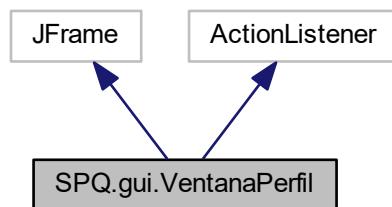
The documentation for this class was generated from the following file:

- `NeverEmptyClient/src/main/java/SPQ/gui/VentanaPago.java`

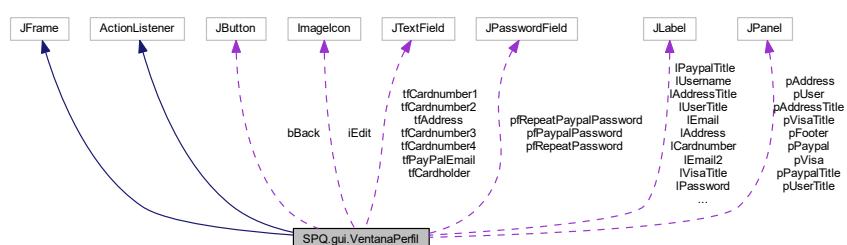
## 7.47 SPQ.gui.VentanaPerfil Class Reference

`VentanaPerfil` is a windows that shows the user profile data. It has 4 sections: NeverEmpty profile data, shipping address, VISA data, PayPal data. On the bottom there is a button that allows the form editing.

Inheritance diagram for `SPQ.gui.VentanaPerfil`:



Collaboration diagram for `SPQ.gui.VentanaPerfil`:



## Public Member Functions

- `VentanaPerfil (NeverEmptyController neverEmptyController, VentanaPrincipal vp)`
- `void passwordModified ()`
- `void actionPerformed (ActionEvent e)`

### 7.47.1 Detailed Description

`VentanaPerfil` is a windows that shows the user profile data. It has 4 sections: NeverEmpty profile data, shipping address, VISA data, PayPal data. On the bottom there is a button that allows the form editing.

Definition at line 33 of file `VentanaPerfil.java`.

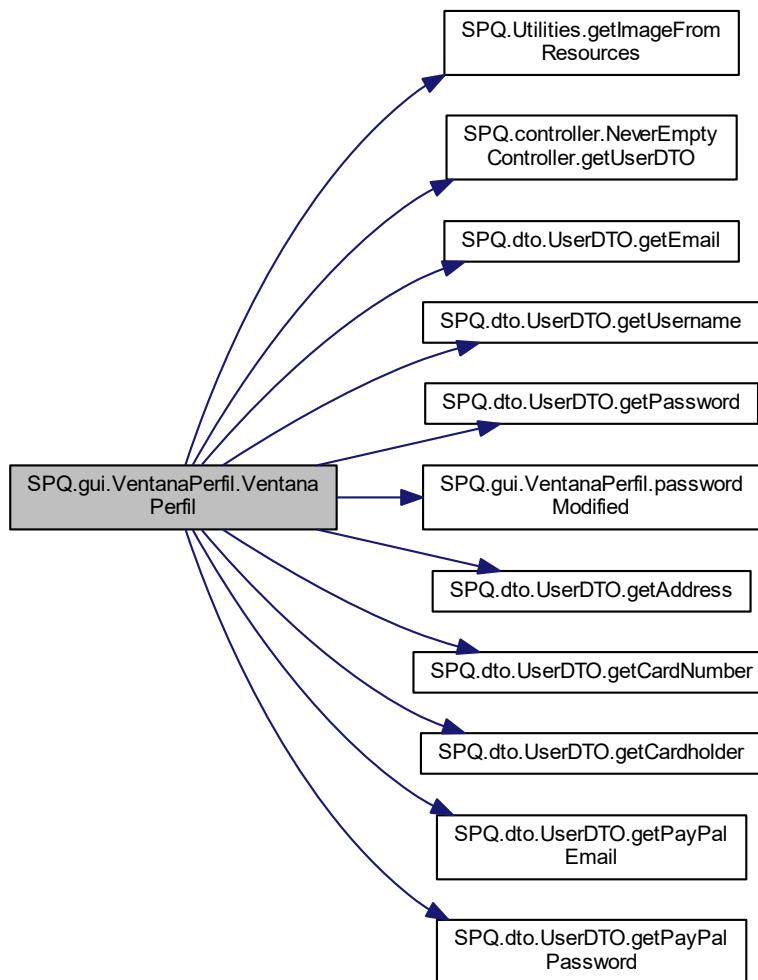
### 7.47.2 Constructor & Destructor Documentation

#### 7.47.2.1 `VentanaPerfil()`

```
SPQ.gui.VentanaPerfil.VentanaPerfil (
    NeverEmptyController neverEmptyController,
    VentanaPrincipal vp )
```

Definition at line 52 of file `VentanaPerfil.java`.

Here is the call graph for this function:



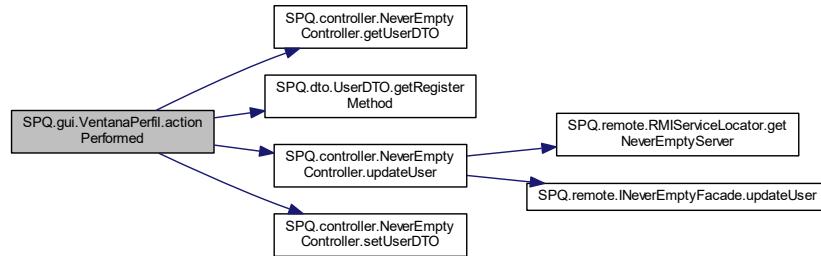
### 7.47.3 Member Function Documentation

#### 7.47.3.1 actionPerformed()

```
void SPQ.gui.VentanaPerfil.actionPerformed ( ActionEvent e )
```

Definition at line 357 of file VentanaPerfil.java.

Here is the call graph for this function:

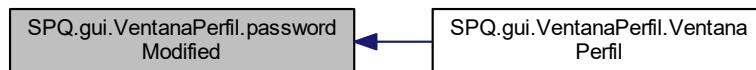


#### 7.47.3.2 passwordModified()

```
void SPQ.gui.VentanaPerfil.passwordModified( )
```

Definition at line 347 of file `VentanaPerfil.java`.

Here is the caller graph for this function:



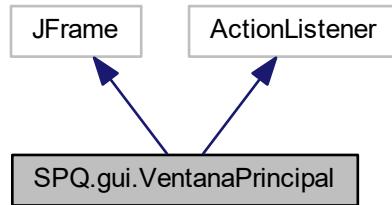
The documentation for this class was generated from the following file:

- NeverEmptyClient/src/main/java/SPQ/gui/[VentanaPerfil.java](#)

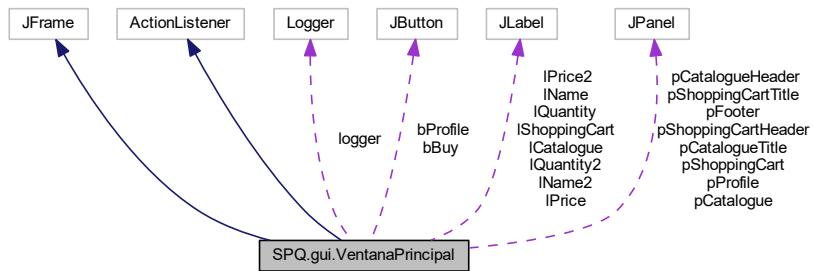
## 7.48 SPQ.gui.VentanaPrincipal Class Reference

This is the main window of the application. It contains two lists: the catalog list with all the products and the shoppingcart list.

Inheritance diagram for SPQ.gui.VentanaPrincipal:



Collaboration diagram for SPQ.gui.VentanaPrincipal:



## Public Member Functions

- [VentanaPrincipal \(NeverEmptyController neverEmptyController\)](#)
- void [actionPerformed \(ActionEvent e\)](#)

## Static Public Member Functions

- static void [main \(String\[ \] args\)](#)

## Public Attributes

- JPanel [pCatalogue](#)

### 7.48.1 Detailed Description

This is the main window of the application. It contains two lists: the catalog list with all the products and the shoppingcart list.

The profile button (bProfile) calls "VentanaPerfil". The sales filter button (bSales) filters the catalog list to show only products with discount. The buy button (bBuy) calls "VentanaTicket".

Definition at line 44 of file VentanaPrincipal.java.

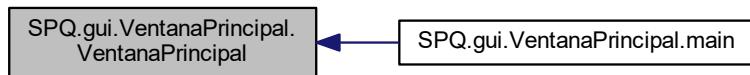
## 7.48.2 Constructor & Destructor Documentation

### 7.48.2.1 VentanaPrincipal()

```
SPQ.gui.VentanaPrincipal.VentanaPrincipal (
    NeverEmptyController neverEmptyController )
```

Definition at line 62 of file VentanaPrincipal.java.

Here is the caller graph for this function:



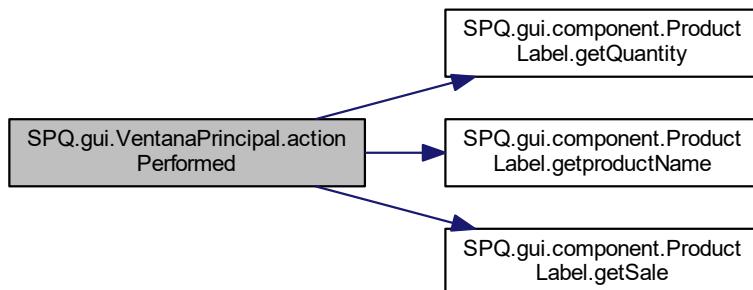
## 7.48.3 Member Function Documentation

### 7.48.3.1 actionPerformed()

```
void SPQ.gui.VentanaPrincipal.actionPerformed (
    ActionEvent e )
```

Definition at line 263 of file VentanaPrincipal.java.

Here is the call graph for this function:

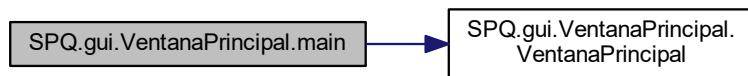


### 7.48.3.2 main()

```
static void SPQ.gui.VentanaPrincipal.main (
    String [] args ) [static]
```

Definition at line 257 of file VentanaPrincipal.java.

Here is the call graph for this function:



### 7.48.4 Member Data Documentation

#### 7.48.4.1 pCatalogue

```
JPanel SPQ.gui.VentanaPrincipal.pCatalogue
```

Definition at line 52 of file VentanaPrincipal.java.

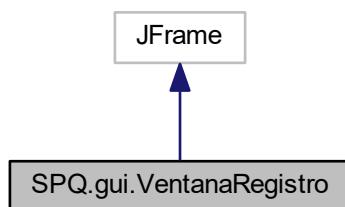
The documentation for this class was generated from the following file:

- NeverEmptyClient/src/main/java/SPQ/gui/[VentanaPrincipal.java](#)

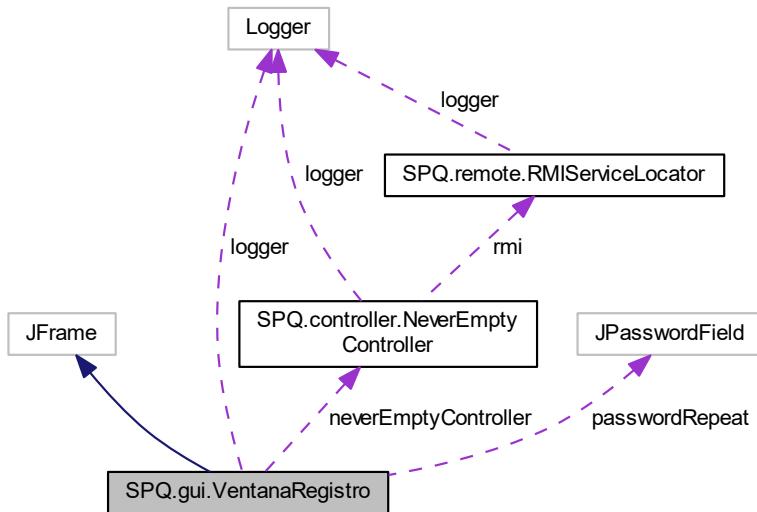
## 7.49 SPQ.gui.VentanaRegistro Class Reference

This window shows the form to signUp on NeverEmpty. The form has 4 fields: username, password, repeat password and email. It is the only information needed to sign up in NeverEmpty. When the register button is pressed the NeverEmptyController login function is called, the form data and the register method is passed to the controller.

Inheritance diagram for SPQ.gui.VentanaRegistro:



Collaboration diagram for SPQ.gui.VentanaRegistro:



## Public Member Functions

- [VentanaRegistro \(NeverEmptyController neverEmptyController, String registerMethod\)](#)
- void [finalizarRegistro \(\)](#)
- boolean [ComprobacionSignup \(\)](#)

## Static Public Member Functions

- static void [main \(String\[\] args\)](#)

### 7.49.1 Detailed Description

This window shows the form to signUp on NeverEmpty. The form has 4 fields: username, password, repeat password and email. It is the only information needed to sign up in NeverEmpty. When the register button is pressed the NeverEmptyController login function is called, the form data and the register method is passed to the controller.

Definition at line 31 of file `VentanaRegistro.java`.

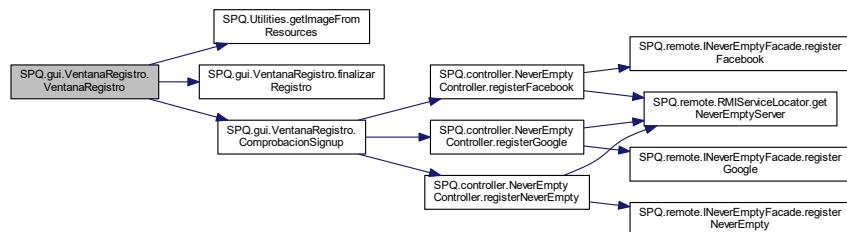
### 7.49.2 Constructor & Destructor Documentation

### 7.49.2.1 VentanaRegistro()

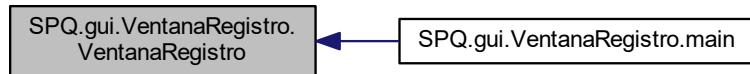
```
SPQ.gui.VentanaRegistro.VentanaRegistro (
    NeverEmptyController neverEmptyController,
    String registerMethod )
```

Definition at line 54 of file VentanaRegistro.java.

Here is the call graph for this function:



Here is the caller graph for this function:



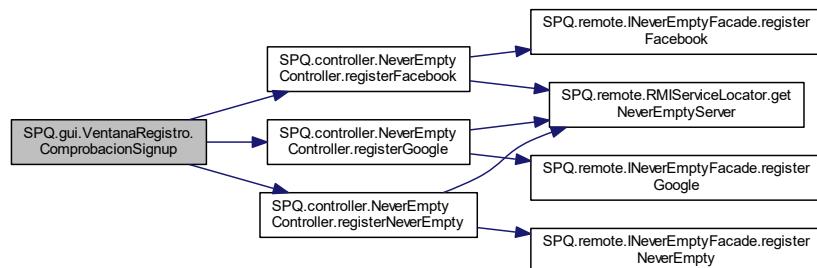
### 7.49.3 Member Function Documentation

#### 7.49.3.1 ComprobacionSignup()

```
boolean SPQ.gui.VentanaRegistro.ComprobacionSignup ( )
```

Definition at line 202 of file VentanaRegistro.java.

Here is the call graph for this function:



Here is the caller graph for this function:

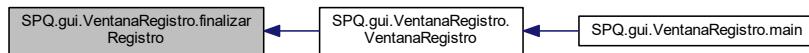


#### 7.49.3.2 finalizarRegistro()

```
void SPQ.gui.VentanaRegistro.finalizarRegistro( )
```

Definition at line 192 of file VentanaRegistro.java.

Here is the caller graph for this function:

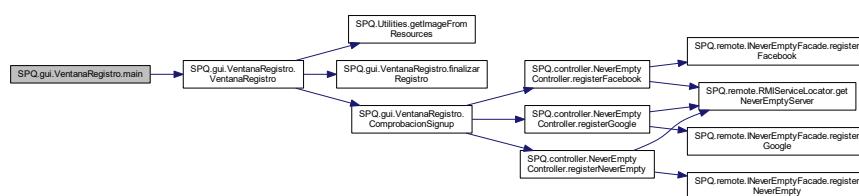


#### 7.49.3.3 main()

```
static void SPQ.gui.VentanaRegistro.main (
    String [] args) [static]
```

Definition at line 50 of file VentanaRegistro.java.

Here is the call graph for this function:



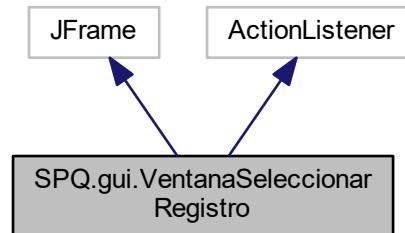
The documentation for this class was generated from the following file:

- NeverEmptyClient/src/main/java/SPQ/gui/[VentanaRegistro.java](#)

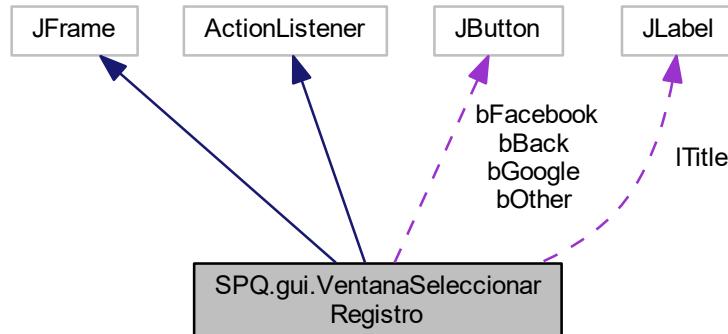
## 7.50 SPQ.gui.VentanaSeleccionarRegistro Class Reference

This window shows the options available to sign up. There are three sign up options: Sign up with google, facebook or NeverEmpty.

Inheritance diagram for SPQ.gui.VentanaSeleccionarRegistro:



Collaboration diagram for SPQ.gui.VentanaSeleccionarRegistro:



### Public Member Functions

- `VentanaSeleccionarRegistro (NeverEmptyController neverEmptyController, VentanalInicio ventanalInicio)`
- `void actionPerformed (ActionEvent e)`

#### 7.50.1 Detailed Description

This window shows the options available to sign up. There are three sign up options: Sign up with google, facebook or NeverEmpty.

Definition at line 26 of file `VentanaSeleccionarRegistro.java`.

## 7.50.2 Constructor & Destructor Documentation

### 7.50.2.1 VentanaSeleccionarRegistro()

```
SPQ.gui.VentanaSeleccionarRegistro.VentanaSeleccionarRegistro (
    NeverEmptyController neverEmptyController,
    VentanaInicio ventanaInicio )
```

Definition at line 42 of file VentanaSeleccionarRegistro.java.

Here is the call graph for this function:



## 7.50.3 Member Function Documentation

### 7.50.3.1 actionPerformed()

```
void SPQ.gui.VentanaSeleccionarRegistro.actionPerformed (
    ActionEvent e )
```

Definition at line 127 of file VentanaSeleccionarRegistro.java.

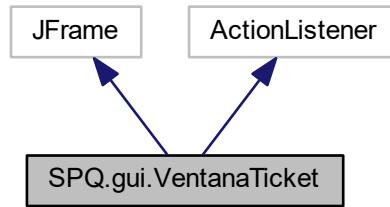
The documentation for this class was generated from the following file:

- NeverEmptyClient/src/main/java/SPQ/gui/[VentanaSeleccionarRegistro.java](#)

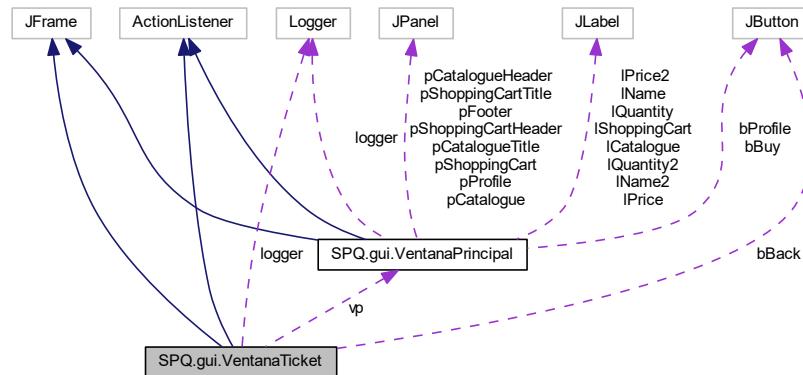
## 7.51 SPQ.gui.VentanaTicket Class Reference

This window shows the list of products added to the shopping list, with its respective prices and the total amount in a table.

Inheritance diagram for SPQ.gui.VentanaTicket:



Collaboration diagram for SPQ.gui.VentanaTicket:



### Public Member Functions

- `VentanaTicket (NeverEmptyController neverEmptyController, neverEmptyController, List< ProductLabel > shoppingList, VentanaPrincipal ventanaPrincipal)`
- void `actionPerformed (ActionEvent e)`

### Static Public Member Functions

- static void `main (String[ ] args)`

## Public Attributes

- `VentanaPrincipal vp`

### 7.51.1 Detailed Description

This window shows the list of products added to the shopping list, with its respective prices and the total amount in a table.

Definition at line 33 of file `VentanaTicket.java`.

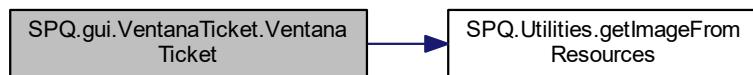
### 7.51.2 Constructor & Destructor Documentation

#### 7.51.2.1 `VentanaTicket()`

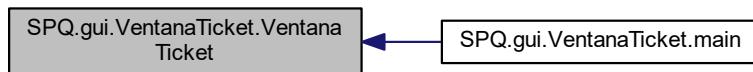
```
SPQ.gui.VentanaTicket.VentanaTicket (
    NeverEmptyController neverEmptyController,
    List< ProductLabel > shoppingList,
    VentanaPrincipal ventanaPrincipal )
```

Definition at line 44 of file `VentanaTicket.java`.

Here is the call graph for this function:



Here is the caller graph for this function:



### 7.51.3 Member Function Documentation

### 7.51.3.1 actionPerformed()

```
void SPQ.gui.VentanaTicket.actionPerformed (
    ActionEvent e )
```

Definition at line 134 of file VentanaTicket.java.

### 7.51.3.2 main()

```
static void SPQ.gui.VentanaTicket.main (
    String [] args ) [static]
```

Definition at line 148 of file VentanaTicket.java.

Here is the call graph for this function:



## 7.51.4 Member Data Documentation

### 7.51.4.1 vp

`VentanaPrincipal` SPQ.gui.VentanaTicket.vp

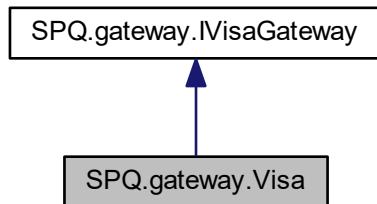
Definition at line 43 of file VentanaTicket.java.

The documentation for this class was generated from the following file:

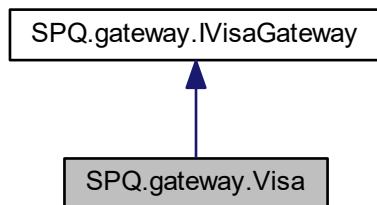
- NeverEmptyClient/src/main/java/SPQ/gui/[VentanaTicket.java](#)

## 7.52 SPQ.gateway.Visa Class Reference

Inheritance diagram for SPQ.gateway.Visa:



Collaboration diagram for SPQ.gateway.Visa:



### Public Member Functions

- [Visa](#) (String serverIP, String serverPort)
- boolean [pay](#) ([PaymentDTO](#) paymentDTO)

#### 7.52.1 Detailed Description

Definition at line 14 of file `Visa.java`.

#### 7.52.2 Constructor & Destructor Documentation

### 7.52.2.1 Visa()

```
SPQ.gateway.Visa.Visa (
    String serverIP,
    String serverPort )
```

Definition at line 20 of file Visa.java.

### 7.52.3 Member Function Documentation

#### 7.52.3.1 pay()

```
boolean SPQ.gateway.Visa.pay (
    PaymentDTO paymentDTO )
```

Implements [SPQ.gateway.IVisaGateway](#).

Definition at line 31 of file Visa.java.

Here is the caller graph for this function:

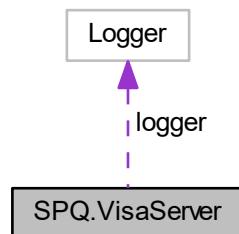


The documentation for this class was generated from the following file:

- [NeverEmptyServer/src/main/java/SPQ/gateway/Visa.java](#)

## 7.53 SPQ.VisaServer Class Reference

Collaboration diagram for SPQ.VisaServer:



## Static Public Member Functions

- static void [main](#) (String args[ ])

### 7.53.1 Detailed Description

Definition at line 8 of file VisaServer.java.

### 7.53.2 Member Function Documentation

#### 7.53.2.1 main()

```
static void SPQ.VisaServer.main (
    String args[ ] ) [static]
```

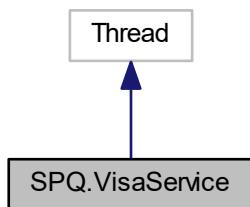
Definition at line 12 of file VisaServer.java.

The documentation for this class was generated from the following file:

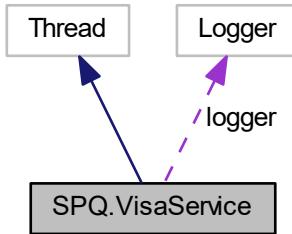
- [VisaService/src/main/java/SPQ/VisaServer.java](#)

## 7.54 SPQ.VisaService Class Reference

Inheritance diagram for SPQ.VisaService:



Collaboration diagram for SPQ.VisaService:



## Public Member Functions

- [VisaService](#) (Socket socket)
- void [run](#) ()
- String [readData](#) (PaymentDTO paymentDTO)

### 7.54.1 Detailed Description

Definition at line 15 of file VisaService.java.

### 7.54.2 Constructor & Destructor Documentation

#### 7.54.2.1 VisaService()

```
SPQ.VisaService.VisaService (
    Socket socket )
```

Definition at line 21 of file VisaService.java.

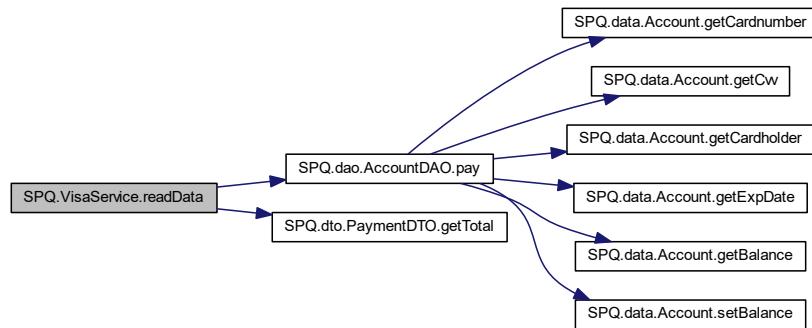
### 7.54.3 Member Function Documentation

### 7.54.3.1 `readData()`

```
String SPQ.VisaService.readData (
    PaymentDTO paymentDTO )
```

Definition at line 54 of file VisaService.java.

Here is the call graph for this function:



Here is the caller graph for this function:

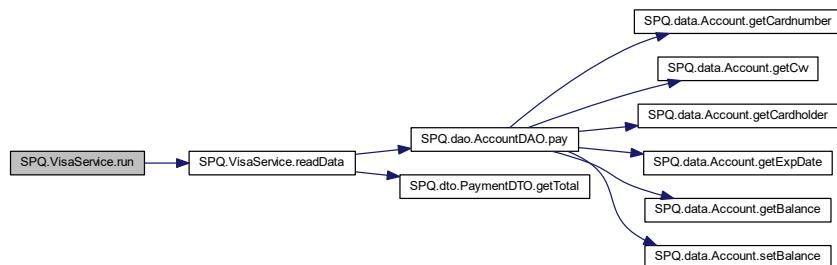


### 7.54.3.2 `run()`

```
void SPQ.VisaService.run ( )
```

Definition at line 32 of file VisaService.java.

Here is the call graph for this function:



The documentation for this class was generated from the following file:

- VisaService/src/main/java/SPQ/[VisaService.java](#)



# Chapter 8

## File Documentation

### 8.1 EroskiService/src/main/java/SPQ/dao/ProductDAO.java File Reference

#### Classes

- class [SPQ.dao.ProductDAO](#)

#### Packages

- package [SPQ.dao](#)

### 8.2 EroskiService/src/main/java/SPQ/data/Product.java File Reference

#### Classes

- class [SPQ.data.Product](#)

#### Packages

- package [SPQ.data](#)

### 8.3 NeverEmptyServer/src/main/java/SPQ/data/Product.java File Reference

#### Classes

- class [SPQ.data.Product](#)

#### Packages

- package [SPQ.data](#)

## 8.4 EroskiService/src/main/java/SPQ/dto/ProductDTO.java File Reference

### Classes

- class [SPQ.dto.ProductDTO](#)

*This class prepares a product to be transferred.*

### Packages

- package [SPQ.dto](#)

## 8.5 NeverEmptyServer/src/main/java/SPQ/dto/ProductDTO.java File Reference

### Classes

- class [SPQ.dto.ProductDTO](#)

*This class prepares a product to be transferred.*

### Packages

- package [SPQ.dto](#)

## 8.6 EroskiService/src/main/java/SPQ/EroskiServer.java File Reference

### Classes

- class [SPQ.EroskiServer](#)

### Packages

- package [SPQ](#)

*Client root package.*

## 8.7 EroskiService/src/main/java/SPQ/EroskiService.java File Reference

### Classes

- class [SPQ.EroskiService](#)

### Packages

- package [SPQ](#)

*Client root package.*

## 8.8 FacebookService/src/main/java/SPQ/dao/IUserDAO.java File Reference

### Classes

- interface [SPQ.dao.IUserDAO](#)

### Packages

- package [SPQ.dao](#)

## 8.9 GoogleService/src/main/java/SPQ/dao/IUserDAO.java File Reference

### Classes

- interface [SPQ.dao.IUserDAO](#)

### Packages

- package [SPQ.dao](#)

## 8.10 PaypalService/src/main/java/SPQ/dao/IUserDAO.java File Reference

### Classes

- interface [SPQ.dao.IUserDAO](#)

### Packages

- package [SPQ.dao](#)

## 8.11 FacebookService/src/main/java/SPQ/dao/UserDAO.java File Reference

### Classes

- class [SPQ.dao.UserDAO](#)

### Packages

- package [SPQ.dao](#)

## 8.12 GoogleService/src/main/java/SPQ/dao/UserDAO.java File Reference

### Classes

- class [SPQ.dao.UserDAO](#)

### Packages

- package [SPQ.dao](#)

## 8.13 NeverEmptyServer/src/main/java/SPQ/dao/UserDAO.java File Reference

### Classes

- class [SPQ.dao.UserDAO](#)

### Packages

- package [SPQ.dao](#)

## 8.14 PaypalService/src/main/java/SPQ/dao/UserDAO.java File Reference

### Classes

- class [SPQ.dao.UserDAO](#)

### Packages

- package [SPQ.dao](#)

## 8.15 FacebookService/src/main/java/SPQ/data/User.java File Reference

### Classes

- class [SPQ.data.User](#)  
*This class defines a [User](#).*

### Packages

- package [SPQ.data](#)

## 8.16 GoogleService/src/main/java/SPQ/data/User.java File Reference

### Classes

- class [SPQ.data.User](#)  
*This class defines a User.*

### Packages

- package [SPQ.data](#)

## 8.17 NeverEmptyServer/src/main/java/SPQ/data/User.java File Reference

### Classes

- class [SPQ.data.User](#)  
*This class defines a User.*

### Packages

- package [SPQ.data](#)

## 8.18 PaypalService/src/main/java/SPQ/data/User.java File Reference

### Classes

- class [SPQ.data.User](#)  
*This class defines a User.*

### Packages

- package [SPQ.data](#)

## 8.19 FacebookService/src/main/java/SPQ/dto/UserDTO.java File Reference

### Classes

- class [SPQ.dto.UserDTO](#)  
*This class prepares a user to be transferred.*

### Packages

- package [SPQ.dto](#)

## 8.20 GoogleService/src/main/java/SPQ/dto/UserDTO.java File Reference

### Classes

- class [SPQ.dto.UserDTO](#)  
*This class prepares a user to be transferred.*

### Packages

- package [SPQ.dto](#)

## 8.21 NeverEmptyServer/src/main/java/SPQ/dto/UserDTO.java File Reference

### Classes

- class [SPQ.dto.UserDTO](#)  
*This class prepares a user to be transferred.*

### Packages

- package [SPQ.dto](#)

## 8.22 FacebookService/src/main/java/SPQ/FacebookServer.java File Reference

### Classes

- class [SPQ.FacebookServer](#)

### Packages

- package [SPQ](#)  
*Client root package.*

## 8.23 FacebookService/src/main/java/SPQ/FacebookService.java File Reference

### Classes

- class [SPQ.FacebookService](#)

### Packages

- package [SPQ](#)  
*Client root package.*

## 8.24 GoogleService/src/main/java/SPQ/GoogleServer.java File Reference

### Classes

- class [SPQ.GoogleServer](#)

### Packages

- package [SPQ](#)

*Client root package.*

## 8.25 GoogleService/src/main/java/SPQ/GoogleService.java File Reference

### Classes

- class [SPQ.GoogleService](#)

### Packages

- package [SPQ](#)

*Client root package.*

## 8.26 NeverEmptyClient/src/main/java/SPQ/controller/NeverEmptyController.java File Reference

### Classes

- class [SPQ.controller.NeverEmptyController](#)

*This class acts as an intermediary between the graphic user interface and the classes that call the server. It mostly contains registration, login and payment methods.*

### Packages

- package [SPQ.controller](#)

*This package contains the classes that act as intermediaries between the graphic user interface and the classes that speak with the server.*

## 8.27 NeverEmptyClient/src/main/java/SPQ/gui/component/ProductLabel.java File Reference

### Classes

- class [SPQ.gui.component.ProductLabel](#)

*This component extends JPanel. Represents one product. [ProductLabel](#) consists of four labels: product name, price, sale (if > 0%), quantity; and two buttons: to increase the quantity by one and to subtract by one.*

## Packages

- package [SPQ.gui.component](#)

*The SPQ.gui.component consist of GUI tiny components that are used by other GUI bigger components as windows.*

## 8.28 NeverEmptyClient/src/main/java/SPQ/gui/Ventanalnicio.java File Reference

### Classes

- class [SPQ.gui.Ventanalnicio](#)

*This is the first window of the application. It contains a form with username and password fields to login. And two buttons: Login and Sign Up. The login button (bLogin) calls the login function of the controller and sends the form data. The sign up (bSignUp) button calls "VentanaSeleccionarRegistro" window.*

## Packages

- package [SPQ.gui](#)

*The SPQ.gui package contains the classes responsible for displaying the graphic user interface of the application.*

## 8.29 NeverEmptyClient/src/main/java/SPQ/gui/VentanaMetodoPago.java File Reference

### Classes

- class [SPQ.gui.VentanaMetodoPago](#)

*This window contains two main buttons, paypal (bPaypal) button and visa button (bVisa). Both of them call "VentanaPago" but bPayPal passes "paypal" String to "VentanaPago" and bVisa "visa".*

## Packages

- package [SPQ.gui](#)

*The SPQ.gui package contains the classes responsible for displaying the graphic user interface of the application.*

## 8.30 NeverEmptyClient/src/main/java/SPQ/gui/VentanaPago.java File Reference

### Classes

- class [SPQ.gui.VentanaPago](#)

*This window shows the last step to make a payment. VentanaPago has two modes: visa and paypal. Depending on the input attribute the form changes. On the top is always shown the shipping address that can be modified in "VentanaPerfil" The PayPal form consist of two fields: username and password. The VISA form consist of four fields: cardnumber, cardholder, expiration date and CVV. The left bottom button allows to use a new paypal user or visa data to pay. It calls "Animation" and transforms the form. The right bottom button calls the NeverEmptyController payWithPayPal(PaymentDTO) or payWithVisa(PaymentDTO) function.*

- class [SPQ.gui.VentanaPago.Animate](#)

## Packages

- package [SPQ.gui](#)

*The SPQ.gui package contains the classes responsible for displaying the graphic user interface of the application.*

## 8.31 NeverEmptyClient/src/main/java/SPQ/gui/VentanaPerfil.java File Reference

### Classes

- class [SPQ.gui.VentanaPerfil](#)

*VentanaPerfil* is a windows that shows the user profile data. It has 4 sections: NeverEmpty profile data, shipping address, VISA data, PayPal data. On the bottom there is a button that allows the form editing.

## Packages

- package [SPQ.gui](#)

*The SPQ.gui package contains the classes responsible for displaying the graphic user interface of the application.*

## 8.32 NeverEmptyClient/src/main/java/SPQ/gui/VentanaPrincipal.java File Reference

### Classes

- class [SPQ.gui.VentanaPrincipal](#)

*This is the main window of the application. It contains two lists: the catalog list with all the products and the shoppingcart list.*

## Packages

- package [SPQ.gui](#)

*The SPQ.gui package contains the classes responsible for displaying the graphic user interface of the application.*

## 8.33 NeverEmptyClient/src/main/java/SPQ/gui/VentanaRegistro.java File Reference

### Classes

- class [SPQ.gui.VentanaRegistro](#)

*This window shows the form to signUp on NeverEmpty. The form has 4 fields: username, password, repeat password and email. It is the only information needed to sign up in NeverEmpty. When the register button is pressed the NeverEmptyController login function is called, the form data and the register method is passed to the controller.*

## Packages

- package [SPQ.gui](#)

*The SPQ.gui package contains the classes responsible for displaying the graphic user interface of the application.*

## 8.34 NeverEmptyClient/src/main/java/SPQ/gui/VentanaSeleccionarRegistro.java File Reference

### Classes

- class [SPQ.gui.VentanaSeleccionarRegistro](#)

*This window shows the options available to sign up. There are three sign up options: Sign up with google, facebook or NeverEmpty.*

### Packages

- package [SPQ.gui](#)

*The SPQ.gui package contains the classes responsible for displaying the graphic user interface of the application.*

## 8.35 NeverEmptyClient/src/main/java/SPQ/gui/VentanaTicket.java File Reference

### Classes

- class [SPQ.gui.VentanaTicket](#)

*This window shows the list of products added to the shopping list, with its respective prices and the total amount in a table.*

### Packages

- package [SPQ.gui](#)

*The SPQ.gui package contains the classes responsible for displaying the graphic user interface of the application.*

## 8.36 NeverEmptyClient/src/main/java/SPQ/main/Main.java File Reference

### Classes

- class [SPQ.main.Main](#)

*The Main class contains the main method of the application, the start point of the client execution.*

### Packages

- package [SPQ.main](#)

*The SPQ.main contains the main class.*

## 8.37 NeverEmptyClient/src/main/java/SPQ/remote/RMIServiceLocator.java File Reference

### Classes

- class [SPQ.remote.RMIServiceLocator](#)

*RMIServiceLocator prepares the connection to the server side by founding the NeverEmptyFacade in the rmi registry.*

## Packages

- package [SPQ.remote](#)

*The [SPQ.remote](#) contains the RMI service locator.*

## 8.38 NeverEmptyClient/src/main/java/SPQ/Utilities.java File Reference

### Classes

- class [SPQ.Utilities](#)

*Utilities class. The [Utilities](#) class gathers reusable utility methods in different points of the project.*

## Packages

- package [SPQ](#)

*Client root package.*

## 8.39 NeverEmptyClient/src/test/java/SPQ/ClientTest.java File Reference

### Classes

- class [SPQ.ClientTest](#)

*ClientTest contains the junit tests of the client side.*

## Packages

- package [SPQ](#)

*Client root package.*

## 8.40 NeverEmptyServer/src/main/java/SPQ/dto/PaymentDTO.java File Reference

### Classes

- class [SPQ.dto.PaymentDTO](#)

*This class prepares a payment to be transferred.*

## Packages

- package [SPQ.dto](#)

## 8.41 PaypalService/src/main/java/SPQ/dto/PaymentDTO.java File Reference

### Classes

- class [SPQ.dto.PaymentDTO](#)

*This class prepares a payment to be transferred.*

### Packages

- package [SPQ.dto](#)

## 8.42 VisaService/src/main/java/SPQ/dto/PaymentDTO.java File Reference

### Classes

- class [SPQ.dto.PaymentDTO](#)

*This class prepares a payment to be transferred.*

### Packages

- package [SPQ.dto](#)

## 8.43 NeverEmptyServer/src/main/java/SPQ/gateway/Eroski.java File Reference

### Classes

- class [SPQ.gateway.Eroski](#)

### Packages

- package [SPQ.gateway](#)

## 8.44 NeverEmptyServer/src/main/java/SPQ/gateway/Facebook.java File Reference

### Classes

- class [SPQ.gateway.Facebook](#)

### Packages

- package [SPQ.gateway](#)

## 8.45 NeverEmptyServer/src/main/java/SPQ/gateway/Google.java File Reference

### Classes

- class [SPQ.gateway.Google](#)

### Packages

- package [SPQ.gateway](#)

## 8.46 NeverEmptyServer/src/main/java/SPQ/gateway/IEroskiGateway.java File Reference

### Classes

- interface [SPQ.gateway.IEroskiGateway](#)

### Packages

- package [SPQ.gateway](#)

## 8.47 NeverEmptyServer/src/main/java/SPQ/gateway/IFacebookGateway.java File Reference

### Classes

- interface [SPQ.gateway.IFacebookGateway](#)

### Packages

- package [SPQ.gateway](#)

## 8.48 NeverEmptyServer/src/main/java/SPQ/gateway/IGoogleGateway.java File Reference

### Classes

- interface [SPQ.gateway.IGoogleGateway](#)

### Packages

- package [SPQ.gateway](#)

## 8.49 NeverEmptyServer/src/main/java/SPQ/gateway/IPayPalGateway.java File Reference

### Classes

- interface [SPQ.gateway.IPayPalGateway](#)

### Packages

- package [SPQ.gateway](#)

## 8.50 NeverEmptyServer/src/main/java/SPQ/gateway/IVisaGateway.java File Reference

### Classes

- interface [SPQ.gateway.IVisaGateway](#)

### Packages

- package [SPQ.gateway](#)

## 8.51 NeverEmptyServer/src/main/java/SPQ/gateway/PayPal.java File Reference

### Classes

- class [SPQ.gateway.PayPal](#)

### Packages

- package [SPQ.gateway](#)

## 8.52 NeverEmptyServer/src/main/java/SPQ/gateway/Visa.java File Reference

### Classes

- class [SPQ.gateway.Visa](#)

### Packages

- package [SPQ.gateway](#)

## 8.53 NeverEmptyServer/src/main/java/SPQ/NeverEmptyServer.java File Reference

### Classes

- class [SPQ.NeverEmptyServer](#)

*This class contains the methods that register and login a user, and recover information about them.*

### Packages

- package [SPQ](#)

*Client root package.*

## 8.54 NeverEmptyServer/src/main/java/SPQ/NeverEmptyServerManager.java File Reference

### Classes

- class [SPQ.NeverEmptyServerManager](#)

*This class contains the main method to execute the server side.*

### Packages

- package [SPQ](#)

*Client root package.*

## 8.55 NeverEmptyServer/src/main/java/SPQ/remote/INeverEmptyFacade.java File Reference

### Classes

- interface [SPQ.remote.INeverEmptyFacade](#)

### Packages

- package [SPQ.remote](#)

*The SPQ.remote contains the RMI service locator.*

## 8.56 NeverEmptyServer/src/main/java/SPQ/remote/NeverEmptyFacade.java File Reference

### Classes

- class [SPQ.remote.NeverEmptyFacade](#)

## Packages

- package [SPQ.remote](#)

*The [SPQ.remote](#) contains the RMI service locator.*

## 8.57 NeverEmptyServer/src/test/java/SPQ/CoberturaTest.java File Reference

### Classes

- class [SPQ.CoberturaTest](#)

### Packages

- package [SPQ](#)

*Client root package.*

## 8.58 NeverEmptyServer/src/test/java/SPQ/DAOMockTest.java File Reference

### Classes

- class [SPQ.DAOMockTest](#)

### Packages

- package [SPQ](#)

*Client root package.*

## 8.59 NeverEmptyServer/src/test/java/SPQ/DATAProductTest.java File Reference

### Classes

- class [SPQ.DATAProductTest](#)

### Packages

- package [SPQ](#)

*Client root package.*

## 8.60 NeverEmptyServer/src/test/java/SPQ/DATAUserTest.java File Reference

### Classes

- class [SPQ.DATAUserTest](#)

## Packages

- package [SPQ](#)

*Client root package.*

## 8.61 NeverEmptyServer/src/test/java/SPQ/RMITest.java File Reference

### Classes

- class [SPQ.RMITest](#)

## Packages

- package [SPQ](#)

*Client root package.*

## 8.62 PaypalService/src/main/java/SPQ/PaypalServer.java File Reference

### Classes

- class [SPQ.PaypalServer](#)

## Packages

- package [SPQ](#)

*Client root package.*

## 8.63 PaypalService/src/main/java/SPQ/PaypalService.java File Reference

### Classes

- class [SPQ.PaypalService](#)

## Packages

- package [SPQ](#)

*Client root package.*

## 8.64 README.md File Reference

## 8.65 VisaService/src/main/java/SPQ/dao/AccountDAO.java File Reference

### Classes

- class [SPQ.dao.AccountDAO](#)

## Packages

- package [SPQ.dao](#)

## 8.66 VisaService/src/main/java/SPQ/dao/IAccountDAO.java File Reference

### Classes

- interface [SPQ.dao.IAccountDAO](#)

## Packages

- package [SPQ.dao](#)

## 8.67 VisaService/src/main/java/SPQ/data/Account.java File Reference

### Classes

- class [SPQ.data.Account](#)

## Packages

- package [SPQ.data](#)

## 8.68 VisaService/src/main/java/SPQ/VisaServer.java File Reference

### Classes

- class [SPQ.VisaServer](#)

## Packages

- package [SPQ](#)

*Client root package.*

## 8.69 VisaService/src/main/java/SPQ/VisaService.java File Reference

### Classes

- class [SPQ.VisaService](#)

## Packages

- package [SPQ](#)

*Client root package.*

# Index

Account  
    SPQ::data::Account, 19

AccountDAO  
    SPQ::dao::AccountDAO, 24

actionPerformed  
    SPQ::gui::VentanaMetodoPago, 186  
    SPQ::gui::VentanaPago, 189  
    SPQ::gui::VentanaPerfil, 193  
    SPQ::gui::VentanaPrincipal, 196  
    SPQ::gui::VentanaSeleccionarRegistro, 202  
    SPQ::gui::VentanaTicket, 204  
    SPQ::gui::component::ProductLabel, 131

Alvarotest  
    SPQ::DATAUserTest, 39

calculateProgress  
    SPQ::gui::VentanaPago, 189, 190

ComprobacionSignup  
    SPQ::gui::VentanaRegistro, 199

Cristiantest  
    SPQ::DATAUserTest, 40

deleteDatabase  
    SPQ::RMITest, 138

deleteProduct  
    SPQ::dao::ProductDAO, 126

Ekaitztest  
    SPQ::DATAUserTest, 41

Enaratest  
    SPQ::DATAUserTest, 42

Eroski  
    SPQ::gateway::Eroski, 45

EroskiService  
    SPQ::EroskiService, 47

EroskiService/src/main/java/SPQ/EroskiServer.java,  
    214

EroskiService/src/main/java/SPQ/EroskiService.java,  
    214

EroskiService/src/main/java/SPQ/dao/ProductDAO.java,  
    213

EroskiService/src/main/java/SPQ/data/Product.java,  
    213

EroskiService/src/main/java/SPQ/dto/ProductDTO.java,  
    214

Facebook  
    SPQ::gateway::Facebook, 50

FacebookService  
    SPQ::FacebookService, 52

FacebookService/src/main/java/SPQ/Facebook↔  
    Server.java, 218

FacebookService/src/main/java/SPQ/Facebook↔  
    Service.java, 218

FacebookService/src/main/java/SPQ/dao/IUserDAO.java,  
    215

FacebookService/src/main/java/SPQ/dao/UserDAO.java,  
    215

FacebookService/src/main/java/SPQ/data/User.java,  
    216

FacebookService/src/main/java/SPQ/dto/UserDTO.java,  
    217

finalizarRegistro  
    SPQ::gui::VentanaRegistro, 200

getAddress  
    SPQ::data::User, 146  
    SPQ::dto::UserDTO, 171

getBalance  
    SPQ::data::Account, 20  
    SPQ::data::User, 146

getCardNumber  
    SPQ::data::User, 147  
    SPQ::dto::UserDTO, 172

getCardholder  
    SPQ::data::Account, 20  
    SPQ::data::User, 146  
    SPQ::dto::PaymentDTO, 104  
    SPQ::dto::UserDTO, 171

getCardnumber  
    SPQ::data::Account, 21  
    SPQ::dto::PaymentDTO, 105

getCvv  
    SPQ::data::Account, 21  
    SPQ::dto::PaymentDTO, 105, 106

getEmail  
    SPQ::data::User, 147, 148  
    SPQ::dto::UserDTO, 173

getExpDate  
    SPQ::data::Account, 21  
    SPQ::dto::PaymentDTO, 106, 107

getImageFromResources  
    SPQ::Utilities, 180

getName  
    SPQ::data::Product, 121

getNeverEmptyServer  
    SPQ::remote::RMIServiceLocator, 135

getPassword  
    SPQ::data::User, 148, 149  
    SPQ::dto::PaymentDTO, 107

SPQ::dto::UserDTO, 174  
 getPayPalEmail  
   SPQ::data::User, 149  
   SPQ::dto::UserDTO, 175  
 getPayPalPassword  
   SPQ::data::User, 150  
   SPQ::dto::UserDTO, 175  
 getPrice  
   SPQ::data::Product, 121  
   SPQ::gui::component::ProductLabel, 132  
 getProductList  
   SPQ::dto::ProductDTO, 128, 129  
 getProducts  
   SPQ::NeverEmptyServer, 92  
   SPQ::controller::NeverEmptyController, 73  
   SPQ::dao::ProductDAO, 126  
   SPQ::gateway::Eroski, 45  
   SPQ::gateway::IEroskiGateway, 61  
   SPQ::remote::INeverEmptyFacade, 64  
   SPQ::remote::NeverEmptyFacade, 85  
 getQuantity  
   SPQ::data::Product, 122  
   SPQ::gui::component::ProductLabel, 132  
 getRegisterMethod  
   SPQ::data::User, 150  
   SPQ::dto::UserDTO, 176  
 getSale  
   SPQ::data::Product, 122  
   SPQ::gui::component::ProductLabel, 133  
 getTotal  
   SPQ::dto::PaymentDTO, 108  
 getUser  
   SPQ::NeverEmptyServer, 92  
   SPQ::controller::NeverEmptyController, 74  
   SPQ::dao::IUserDAO, 68, 69  
   SPQ::dao::UserDAO, 160, 161  
   SPQ::remote::INeverEmptyFacade, 64  
   SPQ::remote::NeverEmptyFacade, 85  
 getUserDTO  
   SPQ::controller::NeverEmptyController, 74  
 getUsername  
   SPQ::data::User, 151  
   SPQ::dto::PaymentDTO, 108, 109  
   SPQ::dto::UserDTO, 176  
 getbAdd  
   SPQ::gui::component::ProductLabel, 131  
 getbSub  
   SPQ::gui::component::ProductLabel, 132  
 getproductName  
   SPQ::gui::component::ProductLabel, 132  
 Google  
   SPQ::gateway::Google, 55  
 GoogleService  
   SPQ::GoogleService, 58  
 GoogleService/src/main/java/SPQ/GoogleServer.java, 219  
 GoogleService/src/main/java/SPQ/GoogleService.java, 219  
 GoogleService/src/main/java/SPQ/IUserDAO.java, 215  
 GoogleService/src/main/java/SPQ/UserDAO.java, 216  
 GoogleService/src/main/java/SPQ/User.java, 217  
 GoogleService/src/main/java/SPQ/dto/UserDTO.java, 218  
 iniciarAplicacion  
   SPQ::gui::VentanaInicio, 183  
 Jesustest  
   SPQ::DATAUserTest, 42  
 login  
   SPQ::NeverEmptyServer, 93  
   SPQ::controller::NeverEmptyController, 75  
   SPQ::remote::INeverEmptyFacade, 64  
   SPQ::remote::NeverEmptyFacade, 85  
 loginExistingUserTest  
   SPQ::RMITest, 138  
 loginNewUserTest  
   SPQ::RMITest, 138  
 main  
   SPQ::EroskiServer, 46  
   SPQ::FacebookServer, 51  
   SPQ::GoogleServer, 57  
   SPQ::NeverEmptyServerManager, 100  
   SPQ::PaypalServer, 116  
   SPQ::VisaServer, 208  
   SPQ::dao::UserDAO, 161  
   SPQ::gui::VentanaInicio, 183  
   SPQ::gui::VentanaPrincipal, 196  
   SPQ::gui::VentanaRegistro, 200  
   SPQ::gui::VentanaTicket, 205  
   SPQ::main::Main, 71  
 NeverEmptyClient/src/main/java/SPQ/Utilities.java, 223  
 NeverEmptyClient/src/main/java/SPQ/controller/←  
   NeverEmptyController.java, 219  
 NeverEmptyClient/src/main/java/SPQ/gui/Ventana←  
   Inicio.java, 220  
 NeverEmptyClient/src/main/java/SPQ/gui/Ventana←  
   MetodoPago.java, 220  
 NeverEmptyClient/src/main/java/SPQ/gui/Ventana←  
   Pago.java, 220  
 NeverEmptyClient/src/main/java/SPQ/gui/Ventana←  
   Perfil.java, 221  
 NeverEmptyClient/src/main/java/SPQ/gui/Ventana←  
   Principal.java, 221  
 NeverEmptyClient/src/main/java/SPQ/gui/Ventana←  
   Registro.java, 221  
 NeverEmptyClient/src/main/java/SPQ/gui/Ventana←  
   SeleccionarRegistro.java, 222  
 NeverEmptyClient/src/main/java/SPQ/gui/Ventana←  
   Ticket.java, 222  
 NeverEmptyClient/src/main/java/SPQ/gui/component/←  
   ProductLabel.java, 219

NeverEmptyClient/src/main/java/SPQ/main/Main.java, 222  
NeverEmptyClient/src/main/java/SPQ/remote/RMIServiceLocator.java, 222  
NeverEmptyClient/src/test/java/SPQ/ClientTest.java, 223  
NeverEmptyController  
    SPQ::controller::NeverEmptyController, 72  
NeverEmptyFacade  
    SPQ::remote::NeverEmptyFacade, 84  
NeverEmptyServer  
    SPQ::NeverEmptyServer, 91  
NeverEmptyServer/src/main/java/SPQ/NeverEmptyServer.java, 227  
NeverEmptyServer/src/main/java/SPQ/NeverEmptyServerManager.java, 227  
NeverEmptyServer/src/main/java/SPQ/dao/UserDAO.java, 216  
NeverEmptyServer/src/main/java/SPQ/data/Product.java, 213  
NeverEmptyServer/src/main/java/SPQ/data/User.java, 217  
NeverEmptyServer/src/main/java/SPQ/dto/PaymentDTO.java, 223  
NeverEmptyServer/src/main/java/SPQ/dto/ProductDTO.java, 214  
NeverEmptyServer/src/main/java/SPQ/dto/UserDTO.java, 218  
NeverEmptyServer/src/main/java/SPQ/gateway/Eroski.java, 224  
NeverEmptyServer/src/main/java/SPQ/gateway/Facebook.java, 224  
NeverEmptyServer/src/main/java/SPQ/gateway/Google.java, 225  
NeverEmptyServer/src/main/java/SPQ/gateway/I  
    EroskiGateway.java, 225  
NeverEmptyServer/src/main/java/SPQ/gateway/I  
    FacebookGateway.java, 225  
NeverEmptyServer/src/main/java/SPQ/gateway/I  
    GoogleGateway.java, 225  
NeverEmptyServer/src/main/java/SPQ/gateway/IPayPalGateway.java, 226  
NeverEmptyServer/src/main/java/SPQ/gateway/IVisaGateway.java, 226  
NeverEmptyServer/src/main/java/SPQ/gateway/PayPal.java, 226  
NeverEmptyServer/src/main/java/SPQ/gateway/Visa.java, 226  
NeverEmptyServer/src/main/java/SPQ/remote/INeverEmptyFacade.java, 227  
NeverEmptyServer/src/main/java/SPQ/remote/NeverEmptyFacade.java, 227  
NeverEmptyServer/src/test/java/SPQ/CoberturaTest.java, 228  
NeverEmptyServer/src/test/java/SPQ/DAOMockTest.java, 228  
NeverEmptyServer/src/test/java/SPQ/DATAProductTest.java, 228  
NeverEmptyServer/src/test/java/SPQ/DATAUserTest.java, 228  
NeverEmptyServer/src/test/java/SPQ/RMITest.java, 229  
pCatalogue  
    SPQ::gui::VentanaPrincipal, 197  
passwordModified  
    SPQ::gui::VentanaPerfil, 194  
pay  
    SPQ::dao::AccountDAO, 24  
    SPQ::dao::IAccountDAO, 60  
    SPQ::dao::IUserDAO, 69  
    SPQ::dao::UserDAO, 161  
    SPQ::gateway::IPayPalGateway, 68  
    SPQ::gateway::IVisaGateway, 70  
    SPQ::gateway::PayPal, 115  
    SPQ::gateway::Visa, 207  
PayPal  
    SPQ::gateway::PayPal, 114  
payWithPayPal  
    SPQ::controller::NeverEmptyController, 76  
payWithPaypal  
    SPQ::NeverEmptyServer, 94  
    SPQ::remote::INeverEmptyFacade, 65  
    SPQ::remote::NeverEmptyFacade, 86  
payWithVisa  
    SPQ::NeverEmptyServer, 95  
    SPQ::controller::NeverEmptyController, 77  
    SPQ::remote::INeverEmptyFacade, 65  
    SPQ::remote::NeverEmptyFacade, 87  
PaymentDTO  
    SPQ::dto::PaymentDTO, 102–104  
PaypalService  
    SPQ::PaypalService, 117  
PaypalService/src/main/java/SPQ/PaypalServer.java, 229  
PaypalService/src/main/java/SPQ/PaypalService.java, 229  
PaypalService/src/main/java/SPQ/dao/IUserDAO.java, 215  
PaypalService/src/main/java/SPQ/dao/UserDAO.java, 216  
PaypalService/src/main/java/SPQ/data/User.java, 217  
PaypalService/src/main/java/SPQ/dto/PaymentDTO.java, 224  
Product  
    SPQ::data::Product, 120  
ProductDAO  
    SPQ::dao::ProductDAO, 126  
ProductDTOTest  
    SPQ::DATAProductTest, 37  
ProductDTO  
    SPQ::dto::ProductDTO, 128  
ProductLabel  
    SPQ::gui::component::ProductLabel, 131  
ProductTest  
    SPQ::DATAProductTest, 37  
README.md, 229

readData  
 SPQ::EroskiService, 48  
 SPQ::FacebookService, 53  
 SPQ::GoogleService, 58  
 SPQ::PaypalService, 117  
 SPQ::VisaService, 209

register  
 SPQ::gateway::Facebook, 50  
 SPQ::gateway::Google, 55  
 SPQ::gateway::IFacebookGateway, 62  
 SPQ::gateway::IGoogleGateway, 62

registerFacebook  
 SPQ::NeverEmptyServer, 96  
 SPQ::controller::NeverEmptyController, 78  
 SPQ::remote::INeverEmptyFacade, 65  
 SPQ::remote::NeverEmptyFacade, 87

registerFacebookExistingUserTest  
 SPQ::RMITest, 138

registerGoogle  
 SPQ::NeverEmptyServer, 97  
 SPQ::controller::NeverEmptyController, 79  
 SPQ::remote::INeverEmptyFacade, 66  
 SPQ::remote::NeverEmptyFacade, 88

registerGoogleExistingUserTest  
 SPQ::RMITest, 139

registerNeverEmpty  
 SPQ::NeverEmptyServer, 97  
 SPQ::controller::NeverEmptyController, 80  
 SPQ::remote::INeverEmptyFacade, 66  
 SPQ::remote::NeverEmptyFacade, 89

registerUserFacebookTest  
 SPQ::RMITest, 139

registerUserGoogleTest  
 SPQ::RMITest, 139

restoreUser  
 SPQ::ClientTest, 26

run  
 SPQ::EroskiService, 48  
 SPQ::FacebookService, 53  
 SPQ::GoogleService, 59  
 SPQ::PaypalService, 118  
 SPQ::VisaService, 210

SPQ.ClientTest, 25  
 SPQ.CoberturaTest, 32  
 SPQ.controller, 14  
 SPQ.controller.NeverEmptyController, 71  
 SPQ.DAOMockTest, 35  
 SPQ.DATAProductTest, 37  
 SPQ.DATAUserTest, 39  
 SPQ.dao, 14  
 SPQ.dao.AccountDAO, 23  
 SPQ.dao.IAccountDAO, 59  
 SPQ.dao.IUserDAO, 68  
 SPQ.dao.ProductDAO, 125  
 SPQ.dao.UserDAO, 158  
 SPQ.data, 15  
 SPQ.data.Account, 19  
 SPQ.data.Product, 119

SPQ.data.User, 142  
 SPQ.dto, 15  
 SPQ.dto.PaymentDTO, 101  
 SPQ.dto.ProductDTO, 127  
 SPQ.dto.UserDTO, 164  
 SPQ.EroskiServer, 46  
 SPQ.EroskiService, 47  
 SPQ.FacebookServer, 51  
 SPQ.FacebookService, 52  
 SPQ.gateway, 15  
 SPQ.gateway.Eroski, 44  
 SPQ.gateway.Facebook, 49  
 SPQ.gateway.Google, 54  
 SPQ.gateway.IEroskiGateway, 60  
 SPQ.gateway.IFacebookGateway, 61  
 SPQ.gateway.IGoogleGateway, 62  
 SPQ.gateway.IPayPalGateway, 67  
 SPQ.gateway.IVisaGateway, 69  
 SPQ.gateway.PayPal, 114  
 SPQ.gateway.Visa, 206  
 SPQ.GoogleServer, 56  
 SPQ.GoogleService, 57  
 SPQ.gui, 16  
 SPQ.gui.component, 16  
 SPQ.gui.component.ProductLabel, 130  
 SPQ.gui.VentanaInicio, 181  
 SPQ.gui.VentanaMetodoPago, 184  
 SPQ.gui.VentanaPago, 187  
 SPQ.gui.VentanaPerfil, 191  
 SPQ.gui.VentanaPrincipal, 194  
 SPQ.gui.VentanaRegistro, 197  
 SPQ.gui.VentanaSeleccionarRegistro, 201  
 SPQ.gui.VentanaTicket, 203  
 SPQ.main, 17  
 SPQ.main.Main, 70  
 SPQ.NeverEmptyServer, 90  
 SPQ.NeverEmptyServerManager, 99  
 SPQ.PaypalServer, 115  
 SPQ.PaypalService, 116  
 SPQ.RMITest, 137  
 SPQ.remote, 17  
 SPQ.remote.INeverEmptyFacade, 63  
 SPQ.remote.NeverEmptyFacade, 83  
 SPQ.remote.RMIServiceLocator, 135  
 SPQ.Utilities, 180  
 SPQ.VisaServer, 207  
 SPQ.VisaService, 208

SPQ::ClientTest  
 restoreUser, 26  
 setUp, 26  
 testGetProducts, 27  
 test GetUser, 27  
 testLogin, 28  
 testLoginFails, 28  
 testPayWithPayPal, 28  
 testPayWithVisa, 29  
 testRegisterUserFacebook, 29  
 testRegisterUserGoogle, 30

testRegisterUserNeverEmpty, 30  
testSetUpBadArguments, 31  
testUpdateUser, 31  
SPQ::CoberturaTest  
test, 33  
SPQ::DAOMockTest  
setUp, 35  
suite, 35  
testRegisterUserAlreadyExists, 36  
testRegisterUserCorrectly, 36  
SPQ::DATAProductTest  
ProductDTOTest, 37  
ProductTest, 37  
setUp, 38  
SPQ::DATAUserTest  
Alvarotest, 39  
Cristiantest, 40  
Ekaitztest, 41  
Enaratest, 42  
Jesustest, 42  
setUp, 43  
UserDTOTest, 43  
SPQ::EroskiServer  
main, 46  
SPQ::EroskiService  
EroskiService, 47  
readData, 48  
run, 48  
SPQ::FacebookServer  
main, 51  
SPQ::FacebookService  
FacebookService, 52  
readData, 53  
run, 53  
SPQ::GoogleServer  
main, 57  
SPQ::GoogleService  
GoogleService, 58  
readData, 58  
run, 59  
SPQ::NeverEmptyServer  
getProducts, 92  
getUser, 92  
login, 93  
NeverEmptyServer, 91  
payWithPaypal, 94  
payWithVisa, 95  
registerFacebook, 96  
registerGoogle, 97  
registerNeverEmpty, 97  
updateUser, 98  
SPQ::NeverEmptyServerManager  
main, 100  
SPQ::PaypalServer  
main, 116  
SPQ::PaypalService  
PaypalService, 117  
readData, 117  
run, 118  
SPQ::RMITest  
deleteDatabase, 138  
loginExistingUserTest, 138  
loginNewUserTest, 138  
registerFacebookExistingUserTest, 138  
registerGoogleExistingUserTest, 139  
registerUserFacebookTest, 139  
registerUserGoogleTest, 139  
setUp, 140  
setUpClient, 140  
suite, 140  
tearDown, 140  
testGetProducts, 141  
testPayWithPayPal, 141  
testPayWithVisa, 141  
SPQ::Utilities  
getImageFromResources, 180  
SPQ::VisaServer  
main, 208  
SPQ::VisaService  
readData, 209  
run, 210  
VisaService, 209  
SPQ::controller::NeverEmptyController  
getProducts, 73  
getUser, 74  
getUserDTO, 74  
login, 75  
NeverEmptyController, 72  
payWithPayPal, 76  
payWithVisa, 77  
registerFacebook, 78  
registerGoogle, 79  
registerNeverEmpty, 80  
setUserDTO, 81  
updateUser, 82  
SPQ::dao::AccountDAO  
AccountDAO, 24  
pay, 24  
storeAccount, 25  
SPQ::dao::IAccountDAO  
pay, 60  
SPQ::dao::IUserDAO  
getUser, 68, 69  
pay, 69  
storeUser, 69  
SPQ::dao::ProductDAO  
deleteProduct, 126  
getProducts, 126  
ProductDAO, 126  
storeProduct, 126  
SPQ::dao::UserDAO  
getUser, 160, 161  
main, 161  
pay, 161  
storeUser, 162, 163  
updateUser, 163

UserDAO, 159, 160  
 SPQ::data::Account  
   Account, 19  
   getBalance, 20  
   getCardholder, 20  
   getCardnumber, 21  
   getCvv, 21  
   getExpDate, 21  
   setBalance, 22  
   setCardholder, 22  
   setCardnumber, 22  
   setCvv, 22  
   setExpDate, 23  
 SPQ::data::Product  
   getName, 121  
   getPrice, 121  
   getQuantity, 122  
   getSale, 122  
   Product, 120  
   setName, 123  
   setPrice, 123, 124  
   setQuantity, 124  
   setSale, 124, 125  
 SPQ::data::User  
   getAddress, 146  
   getBalance, 146  
   getCardNumber, 147  
   getCardholder, 146  
   getEmail, 147, 148  
   getPassword, 148, 149  
   getPayPalEmail, 149  
   getPayPalPassword, 150  
   getRegisterMethod, 150  
   getUsername, 151  
   setAddress, 151  
   setBalance, 152  
   setCardNumber, 152  
   setCardholder, 152  
   setEmail, 153, 154  
   setPassword, 154, 155  
   setPayPalEmail, 155  
   setPayPalPassword, 155  
   setRegisterMethod, 156  
   setUsername, 156  
   updateUser, 157  
   User, 143–145  
 SPQ::dto::PaymentDTO  
   getCardholder, 104  
   getCardnumber, 105  
   getCvv, 105, 106  
   getExpDate, 106, 107  
   getPassword, 107  
   getTotal, 108  
   getUsername, 108, 109  
   PaymentDTO, 102–104  
   setCardholder, 109, 110  
   setCardnumber, 110  
   setCvv, 110, 111  
   setExpDate, 111  
   setPassword, 112  
   setTotal, 112, 113  
   setUsername, 113  
 SPQ::dto::ProductDTO  
   getProductList, 128, 129  
   ProductDTO, 128  
   setProductList, 129  
 SPQ::dto::UserDTO  
   getAddress, 171  
   getCardNumber, 172  
   getCardholder, 171  
   getEmail, 173  
   getPassword, 174  
   getPayPalEmail, 175  
   getPayPalPassword, 175  
   getRegisterMethod, 176  
   getUsername, 176  
   setAddress, 177  
   setCardNumber, 177  
   setCardholder, 177  
   setEmail, 178  
   setPassword, 178, 179  
   setPayPalEmail, 179  
   setPayPalPassword, 179  
   setRegisterMethod, 179  
   setUsername, 179  
   UserDTO, 165–170  
 SPQ::gateway::Eroski  
   Eroski, 45  
   getProducts, 45  
 SPQ::gateway::Facebook  
   Facebook, 50  
   register, 50  
 SPQ::gateway::Google  
   Google, 55  
   register, 55  
 SPQ::gateway::IEroskiGateway  
   getProducts, 61  
 SPQ::gateway::IFacebookGateway  
   register, 62  
 SPQ::gateway::IGoogleGateway  
   register, 62  
 SPQ::gateway::IPayPalGateway  
   pay, 68  
 SPQ::gateway::IVisaGateway  
   pay, 70  
 SPQ::gateway::PayPal  
   pay, 115  
   PayPal, 114  
 SPQ::gateway::Visa  
   pay, 207  
   Visa, 206  
 SPQ::gui::VentanaInicio  
   iniciarAplicacion, 183  
   main, 183  
   VentanaInicio, 182  
 SPQ::gui::VentanaMetodoPago

actionPerformed, 186  
Ventana MetodoPago, 185  
vt, 186

SPQ::gui::VentanaPago  
actionPerformed, 189  
calculateProgress, 189, 190  
VentanaPago, 188  
vmp, 191

SPQ::gui::VentanaPerfil  
actionPerformed, 193  
passwordModified, 194  
VentanaPerfil, 192

SPQ::gui::VentanaPrincipal  
actionPerformed, 196  
main, 196  
pCatalogue, 197  
VentanaPrincipal, 196

SPQ::gui::VentanaRegistro  
ComprobacionSignup, 199  
finalizarRegistro, 200  
main, 200  
VentanaRegistro, 198

SPQ::gui::VentanaSeleccionarRegistro  
actionPerformed, 202  
VentanaSeleccionarRegistro, 202

SPQ::gui::VentanaTicket  
actionPerformed, 204  
main, 205  
VentanaTicket, 204  
vp, 205

SPQ::gui::component::ProductLabel  
actionPerformed, 131  
getPrice, 132  
getQuantity, 132  
getSale, 133  
getbAdd, 131  
getbSub, 132  
getproductName, 132  
ProductLabel, 131  
setPrice, 134  
setQuantity, 134  
setSale, 134  
setbAdd, 133  
setbSub, 133  
setproductName, 134

SPQ::main::Main  
main, 71

SPQ::remote::INeverEmptyFacade  
getProducts, 64  
getUser, 64  
login, 64  
payWithPaypal, 65  
payWithVisa, 65  
registerFacebook, 65  
registerGoogle, 66  
registerNeverEmpty, 66  
updateUser, 66

SPQ::remote::NeverEmptyFacade  
getProducts, 85  
getUser, 85  
login, 85  
NeverEmptyFacade, 84  
payWithPaypal, 86  
payWithVisa, 87  
registerFacebook, 87  
registerGoogle, 88  
registerNeverEmpty, 89  
updateUser, 89

SPQ::remote::RMIServiceLocator  
getNeverEmptyServer, 135  
setService, 136

SPQ, 13

setAddress  
SPQ::data::User, 151  
SPQ::dto::UserDTO, 177

setBalance  
SPQ::data::Account, 22  
SPQ::data::User, 152

setCardNumber  
SPQ::data::User, 152  
SPQ::dto::UserDTO, 177

setCardholder  
SPQ::data::Account, 22  
SPQ::data::User, 152  
SPQ::dto::PaymentDTO, 109, 110  
SPQ::dto::UserDTO, 177

setCardnumber  
SPQ::data::Account, 22  
SPQ::dto::PaymentDTO, 110

setCvv  
SPQ::data::Account, 22  
SPQ::dto::PaymentDTO, 110, 111

setEmail  
SPQ::data::User, 153, 154  
SPQ::dto::UserDTO, 178

setExpDate  
SPQ::data::Account, 23  
SPQ::dto::PaymentDTO, 111

setName  
SPQ::data::Product, 123

setPassword  
SPQ::data::User, 154, 155  
SPQ::dto::PaymentDTO, 112  
SPQ::dto::UserDTO, 178, 179

setPayPalEmail  
SPQ::data::User, 155  
SPQ::dto::UserDTO, 179

setPayPalPassword  
SPQ::data::User, 155  
SPQ::dto::UserDTO, 179

setPrice  
SPQ::data::Product, 123, 124  
SPQ::gui::component::ProductLabel, 134

setProductList  
SPQ::dto::ProductDTO, 129

setQuantity

SPQ::data::Product, 124  
 SPQ::gui::component::ProductLabel, 134  
**setRegisterMethod**  
 SPQ::data::User, 156  
 SPQ::dto::UserDTO, 179  
**setSale**  
 SPQ::data::Product, 124, 125  
 SPQ::gui::component::ProductLabel, 134  
**setService**  
 SPQ::remote::RMIServiceLocator, 136  
**setTotal**  
 SPQ::dto::PaymentDTO, 112, 113  
**setUp**  
 SPQ::ClientTest, 26  
 SPQ::DAOMockTest, 35  
 SPQ::DATAProductTest, 38  
 SPQ::DATAUserTest, 43  
 SPQ::RMITest, 140  
**setUpClient**  
 SPQ::RMITest, 140  
**setUserDTO**  
 SPQ::controller::NeverEmptyController, 81  
**setUsername**  
 SPQ::data::User, 156  
 SPQ::dto::PaymentDTO, 113  
 SPQ::dto::UserDTO, 179  
**setbAdd**  
 SPQ::gui::component::ProductLabel, 133  
**setbSub**  
 SPQ::gui::component::ProductLabel, 133  
**setproductName**  
 SPQ::gui::component::ProductLabel, 134  
**storeAccount**  
 SPQ::dao::AccountDAO, 25  
**storeProduct**  
 SPQ::dao::ProductDAO, 126  
**storeUser**  
 SPQ::dao::IUserDAO, 69  
 SPQ::dao::UserDAO, 162, 163  
**suite**  
 SPQ::DAOMockTest, 35  
 SPQ::RMITest, 140  
**tearDown**  
 SPQ::RMITest, 140  
**test**  
 SPQ::CoberturaTest, 33  
**testGetProducts**  
 SPQ::ClientTest, 27  
 SPQ::RMITest, 141  
**test GetUser**  
 SPQ::ClientTest, 27  
**testLogin**  
 SPQ::ClientTest, 28  
**testLoginFails**  
 SPQ::ClientTest, 28  
**testPayWithPayPal**  
 SPQ::ClientTest, 28  
 SPQ::RMITest, 141  
**testPayWithVisa**  
 SPQ::ClientTest, 29  
 SPQ::RMITest, 141  
**testRegisterUserAlreadyExists**  
 SPQ::DAOMockTest, 36  
**testRegisterUserCorrectly**  
 SPQ::DAOMockTest, 36  
**testRegisterUserFacebook**  
 SPQ::ClientTest, 29  
**testRegisterUserGoogle**  
 SPQ::ClientTest, 30  
**testRegisterUserNeverEmpty**  
 SPQ::ClientTest, 30  
**testSetUpBadArguments**  
 SPQ::ClientTest, 31  
**testUpdateUser**  
 SPQ::ClientTest, 31  
**updateUser**  
 SPQ::NeverEmptyServer, 98  
 SPQ::controller::NeverEmptyController, 82  
 SPQ::dao::UserDAO, 163  
 SPQ::data::User, 157  
 SPQ::remote::INeverEmptyFacade, 66  
 SPQ::remote::NeverEmptyFacade, 89  
**User**  
 SPQ::data::User, 143–145  
**UserDAO**  
 SPQ::dao::UserDAO, 159, 160  
**UserDTOTest**  
 SPQ::DATAUserTest, 43  
**UserDTO**  
 SPQ::dto::UserDTO, 165–170  
**VentanaInicio**  
 SPQ::gui::VentanaInicio, 182  
**VentanaMetodoPago**  
 SPQ::gui::VentanaMetodoPago, 185  
**VentanaPago**  
 SPQ::gui::VentanaPago, 188  
**VentanaPerfil**  
 SPQ::gui::VentanaPerfil, 192  
**VentanaPrincipal**  
 SPQ::gui::VentanaPrincipal, 196  
**VentanaRegistro**  
 SPQ::gui::VentanaRegistro, 198  
**VentanaSeleccionarRegistro**  
 SPQ::gui::VentanaSeleccionarRegistro, 202  
**VentanaTicket**  
 SPQ::gui::VentanaTicket, 204  
**Visa**  
 SPQ::gateway::Visa, 206  
**VisaService**  
 SPQ::VisaService, 209  
 VisaService/src/main/java/SPQ/VisaServer.java, 230  
 VisaService/src/main/java/SPQ/VisaService.java, 230  
 VisaService/src/main/java/SPQ/dao/AccountDAO.java,  
 229

VisaService/src/main/java/SPQ/dao/IAccountDAO.java,  
[230](#)

VisaService/src/main/java/SPQ/data/Account.java, [230](#)  
VisaService/src/main/java/SPQ/dto/PaymentDTO.java,  
[224](#)

vmp  
    SPQ::gui::VentanaPago, [191](#)

vp  
    SPQ::gui::VentanaTicket, [205](#)

vt  
    SPQ::gui::VentanaMetodoPago, [186](#)