

EventFinder

Generated by Doxygen 1.8.18

1 Deprecated List	1
2 Namespace Index	3
2.1 Packages	3
3 Hierarchical Index	5
3.1 Class Hierarchy	5
4 Class Index	7
4.1 Class List	7
5 File Index	11
5.1 File List	11
6 Namespace Documentation	13
6.1 Package es	13
6.2 Package es.deusto	13
6.3 Package es.deusto.client	13
6.4 Package es.deusto.client.controller	13
6.5 Package es.deusto.client.main	14
6.6 Package es.deusto.client.windows	14
6.7 Package es.deusto.junit	15
6.8 Package es.deusto.junit.client	15
6.9 Package es.deusto.junit.serialization	15
6.10 Package es.deusto.junit.server	15
6.11 Package es.deusto.serialization	15
6.11.1 Detailed Description	16
6.12 Package es.deusto.server	16
6.13 Package es.deusto.server.dao	16
6.14 Package es.deusto.server.data	16
6.15 Package es.deusto.server.server	17
7 Class Documentation	19
7.1 es.deusto.server.server.AppService Class Reference	19
7.1.1 Constructor & Destructor Documentation	20
7.1.1.1 AppService()	20
7.1.2 Member Function Documentation	20
7.1.2.1 attemptNormalLogin()	20
7.1.2.2 attemptNormalSignup()	20
7.1.2.3 attemptNormalUpdate()	21
7.1.2.4 attemptOrganizerLogin()	21
7.1.2.5 attemptOrganizerSignup()	22
7.1.2.6 attemptOrganizerUpdate()	22
7.1.2.7 createEvent()	23

7.1.2.8 createPost()	23
7.1.2.9 deleteOrganizer()	24
7.1.2.10 deleteUser()	24
7.1.2.11 getRecommendedEvents()	25
7.1.2.12 hello()	25
7.1.2.13 recoverPassword()	26
7.1.2.14 updateUser()	26
7.2 es.deusto.client.windows.ChangePass Class Reference	26
7.2.1 Detailed Description	27
7.2.2 Constructor & Destructor Documentation	27
7.2.2.1 ChangePass()	27
7.3 es.deusto.client.windows.CodeVerification Class Reference	27
7.3.1 Constructor & Destructor Documentation	27
7.3.1.1 CodeVerification()	28
7.4 es.deusto.client.controller.Controller Class Reference	28
7.4.1 Detailed Description	29
7.4.2 Constructor & Destructor Documentation	29
7.4.2.1 Controller() [1/2]	29
7.4.2.2 Controller() [2/2]	29
7.4.3 Member Function Documentation	29
7.4.3.1 attemptNormalLogin() [1/2]	30
7.4.3.2 attemptNormalLogin() [2/2]	30
7.4.3.3 attemptNormalLoginOrganizer()	31
7.4.3.4 attemptNormalSignup()	31
7.4.3.5 attemptNormalUpdate()	32
7.4.3.6 attemptOrganizerDelete()	32
7.4.3.7 attemptOrganizerSignup()	33
7.4.3.8 attemptOrganizerUpdate()	33
7.4.3.9 attemptUserDelete()	34
7.4.3.10 createEvent()	34
7.4.3.11 createPost()	34
7.4.3.12 getLanguageManager()	35
7.4.3.13 getOrganize()	35
7.4.3.14 getRecommendations()	35
7.4.3.15 getUser()	36
7.4.3.16 setOrganizer()	36
7.4.3.17 setUser()	36
7.5 es.deusto.junit.client.ControllerTest Class Reference	36
7.5.1 Detailed Description	37
7.5.2 Member Function Documentation	37
7.5.2.1 configure()	37
7.5.2.2 setUpChild()	37

7.5.2.3 startUp()	37
7.5.2.4 suite()	37
7.5.2.5 tearDownChild()	38
7.5.2.6 testNormalLogin()	38
7.5.2.7 testNormalSignupAndDelete()	38
7.5.2.8 testOrganizerLogin()	38
7.5.2.9 testOrganizerSignupAndDelete()	38
7.5.2.10 testUserUpdate()	38
7.5.3 Member Data Documentation	38
7.5.3.1 rule	38
7.6 es.deusto.client.windows.CreateEvent Class Reference	39
7.6.1 Detailed Description	39
7.6.2 Constructor & Destructor Documentation	39
7.6.2.1 CreateEvent()	39
7.6.3 Member Function Documentation	39
7.6.3.1 actionPerformed()	39
7.7 es.deusto.client.windows.CreateOrganizerWindow Class Reference	40
7.7.1 Detailed Description	40
7.7.2 Constructor & Destructor Documentation	40
7.7.2.1 CreateOrganizerWindow()	40
7.7.3 Member Function Documentation	40
7.7.3.1 actionPerformed()	40
7.8 es.deusto.client.windows.CreateUserWindow Class Reference	41
7.8.1 Detailed Description	41
7.8.2 Constructor & Destructor Documentation	41
7.8.2.1 CreateUserWindow()	41
7.8.3 Member Function Documentation	41
7.8.3.1 actionPerformed()	41
7.9 es.deusto.server.dao.DAOFactory Class Reference	42
7.9.1 Detailed Description	42
7.9.2 Member Function Documentation	42
7.9.2.1 closeDAO() [1/5]	42
7.9.2.2 closeDAO() [2/5]	42
7.9.2.3 closeDAO() [3/5]	43
7.9.2.4 closeDAO() [4/5]	43
7.9.2.5 closeDAO() [5/5]	43
7.9.2.6 createEventDAO()	43
7.9.2.7 createOrganizerDAO()	43
7.9.2.8 createPostDAO()	43
7.9.2.9 createTopicDAO()	43
7.9.2.10 createUserDAO()	44
7.9.2.11 getInstance()	44

7.10 es.deusto.server.data.Event Class Reference	44
7.10.1 Detailed Description	44
7.10.2 Constructor & Destructor Documentation	45
7.10.2.1 Event() [1/2]	45
7.10.2.2 Event() [2/2]	45
7.10.3 Member Function Documentation	45
7.10.3.1 addInterested()	45
7.10.3.2 getCity()	45
7.10.3.3 getCountry()	45
7.10.3.4 getDate()	45
7.10.3.5 getDescription()	46
7.10.3.6 getInterested()	46
7.10.3.7 getName()	46
7.10.3.8 getOrganizer()	46
7.10.3.9 getPosts()	46
7.10.3.10 getTopic()	46
7.10.3.11 reduceInterested()	46
7.10.3.12 setCity()	46
7.10.3.13 setCountry()	47
7.10.3.14 setDate()	47
7.10.3.15 setDescription()	47
7.10.3.16 setInterested()	47
7.10.3.17 setName()	47
7.10.3.18 setOrganizer()	47
7.10.3.19 setPosts()	47
7.10.3.20 setTopic()	48
7.10.3.21 toString()	48
7.11 es.deusto.server.dao.EventDAO Class Reference	48
7.11.1 Detailed Description	48
7.11.2 Constructor & Destructor Documentation	49
7.11.2.1 EventDAO()	49
7.11.3 Member Function Documentation	49
7.11.3.1 close()	49
7.11.3.2 deleteEvent()	49
7.11.3.3 getEvents() [1/2]	49
7.11.3.4 getEvents() [2/2]	49
7.11.3.5 getEventsByOrganizer()	50
7.11.3.6 getEventsbyUser()	50
7.11.3.7 getPersistenceManager()	51
7.11.3.8 storeEvent()	51
7.11.3.9 updateEvent()	51
7.12 es.deusto.serialization.EventInfo Class Reference	51

7.12.1 Detailed Description	52
7.12.2 Constructor & Destructor Documentation	52
7.12.2.1 EventInfo() [1/2]	52
7.12.2.2 EventInfo() [2/2]	52
7.12.3 Member Function Documentation	52
7.12.3.1 addInterested()	53
7.12.3.2 getCity()	53
7.12.3.3 getCountry()	53
7.12.3.4 getDate()	53
7.12.3.5 getDescription()	53
7.12.3.6 getInterested()	53
7.12.3.7 getName()	53
7.12.3.8 getOrganizerEmail()	53
7.12.3.9 getPosts()	54
7.12.3.10 getTopic()	54
7.12.3.11 reduceInterested()	54
7.12.3.12 setCity()	54
7.12.3.13 setCountry()	54
7.12.3.14 setDate()	54
7.12.3.15 setDescription()	54
7.12.3.16 setInterested()	55
7.12.3.17 setName()	55
7.12.3.18 setOrganizerEmail()	55
7.12.3.19 setPosts()	55
7.12.3.20 setTopic()	55
7.13 es.deusto.client.windows.EventListItem Class Reference	55
7.13.1 Detailed Description	56
7.13.2 Constructor & Destructor Documentation	56
7.13.2.1 EventListItem()	56
7.14 es.deusto.client.windows.EventOrganizerWindow Class Reference	56
7.14.1 Detailed Description	56
7.14.2 Constructor & Destructor Documentation	57
7.14.2.1 EventOrganizerWindow()	57
7.15 es.deusto.client.windows.EventWindow Class Reference	57
7.15.1 Detailed Description	57
7.15.2 Constructor & Destructor Documentation	57
7.15.2.1 EventWindow()	57
7.16 es.deusto.client.windows.ForgottenPassword Class Reference	58
7.16.1 Detailed Description	58
7.16.2 Constructor & Destructor Documentation	58
7.16.2.1 ForgottenPassword()	58
7.17 es.deusto.server.data.GenericUser Class Reference	58

7.17.1 Detailed Description	59
7.17.2 Member Function Documentation	59
7.17.2.1 getEmail()	59
7.17.2.2 getName()	59
7.17.2.3 getPassword()	59
7.17.2.4 setEmail()	59
7.17.2.5 setName()	60
7.17.2.6 setPassword()	60
7.17.2.7 toString()	60
7.18 es.deusto.serialization.GenericUserInfo Class Reference	60
7.18.1 Detailed Description	60
7.19 es.deusto.client.windows.LanguageManager Class Reference	61
7.19.1 Detailed Description	61
7.19.2 Constructor & Destructor Documentation	61
7.19.2.1 LanguageManager() [1/2]	61
7.19.2.2 LanguageManager() [2/2]	61
7.19.3 Member Function Documentation	62
7.19.3.1 getLanguage()	62
7.19.3.2 getString()	62
7.19.3.3 setLanguage()	62
7.20 es.deusto.serialization.LoginAttempt Class Reference	62
7.20.1 Detailed Description	63
7.20.2 Constructor & Destructor Documentation	63
7.20.2.1 LoginAttempt() [1/2]	63
7.20.2.2 LoginAttempt() [2/2]	63
7.20.3 Member Function Documentation	64
7.20.3.1 getEmail()	64
7.20.3.2 getPassword()	64
7.20.3.3 isOrganizer()	64
7.20.3.4 setEmail()	64
7.20.3.5 setOrganizer()	64
7.20.3.6 setPassword()	64
7.20.3.7 toString()	65
7.21 es.deusto.client.windows.LogInWindow Class Reference	65
7.21.1 Detailed Description	65
7.21.2 Constructor & Destructor Documentation	65
7.21.2.1 LogInWindow()	65
7.21.3 Member Data Documentation	66
7.21.3.1 bCreate	66
7.21.3.2 cBoxAdmin	66
7.21.3.3 id	66
7.22 es.deusto.client.main.Main Class Reference	66

7.22.1 Detailed Description	66
7.22.2 Member Function Documentation	66
7.22.2.1 main()	67
7.23 es.deusto.server.data.Organizer Class Reference	67
7.23.1 Detailed Description	67
7.23.2 Constructor & Destructor Documentation	67
7.23.2.1 Organizer()	67
7.23.3 Member Function Documentation	68
7.23.3.1 addCreatedEvent()	68
7.23.3.2 getCreatedEvents()	68
7.23.3.3 getOrganization()	68
7.23.3.4 setCreatedEvents()	68
7.23.3.5 setOrganization()	68
7.23.3.6 toString()	68
7.24 es.deusto.server.dao.OrganizerDAO Class Reference	69
7.24.1 Detailed Description	69
7.24.2 Constructor & Destructor Documentation	69
7.24.2.1 OrganizerDAO()	69
7.24.3 Member Function Documentation	69
7.24.3.1 close()	69
7.24.3.2 deleteOrganizer()	70
7.24.3.3 getOrganizer()	70
7.24.3.4 getPersistenceManager()	70
7.24.3.5 storeOrganizer()	70
7.24.3.6 updateOrganizer()	70
7.25 es.deusto.client.windows.OrganizerEvents Class Reference	70
7.25.1 Detailed Description	71
7.25.2 Constructor & Destructor Documentation	71
7.25.2.1 OrganizerEvents()	71
7.26 es.deusto.client.windows.OrganizerHome Class Reference	71
7.26.1 Detailed Description	71
7.26.2 Constructor & Destructor Documentation	72
7.26.2.1 OrganizerHome()	72
7.27 es.deusto.serialization.OrganizerInfo Class Reference	72
7.27.1 Detailed Description	72
7.27.2 Constructor & Destructor Documentation	73
7.27.2.1 OrganizerInfo() [1/2]	73
7.27.2.2 OrganizerInfo() [2/2]	73
7.27.3 Member Function Documentation	73
7.27.3.1 getCreatedEvents()	73
7.27.3.2 getEmail()	73
7.27.3.3 getName()	73

7.27.3.4 getOrganization()	73
7.27.3.5 setCreatedEvents()	74
7.27.3.6 setEmail()	74
7.27.3.7 setName()	74
7.27.3.8 setOrganization()	74
7.28 es.deusto.server.data.Post Class Reference	74
7.28.1 Detailed Description	75
7.28.2 Constructor & Destructor Documentation	75
7.28.2.1 Post() [1/2]	75
7.28.2.2 Post() [2/2]	75
7.28.3 Member Function Documentation	75
7.28.3.1 getDate()	75
7.28.3.2 getDescription()	75
7.28.3.3 getEventName()	75
7.28.3.4 getOrganizerEmail()	76
7.28.3.5 getTitle()	76
7.28.3.6 setDate()	76
7.28.3.7 setDescription()	76
7.28.3.8 setEventName()	76
7.28.3.9 setOrganizerEmail()	76
7.28.3.10 setTitle()	76
7.29 es.deusto.client.windows.PostCreationWindow Class Reference	77
7.29.1 Detailed Description	77
7.29.2 Constructor & Destructor Documentation	77
7.29.2.1 PostCreationWindow()	77
7.30 es.deusto.server.dao.PostDAO Class Reference	77
7.30.1 Detailed Description	78
7.30.2 Constructor & Destructor Documentation	78
7.30.2.1 PostDAO()	78
7.30.3 Member Function Documentation	78
7.30.3.1 close()	78
7.30.3.2 deletePost()	78
7.30.3.3 getPersistenceManager()	78
7.30.3.4 getPostsByEvent()	79
7.30.3.5 storePost()	79
7.30.3.6 updatePost()	79
7.31 es.deusto.serialization.PostInfo Class Reference	79
7.31.1 Detailed Description	79
7.31.2 Constructor & Destructor Documentation	80
7.31.2.1 PostInfo() [1/2]	80
7.31.2.2 PostInfo() [2/2]	80
7.31.3 Member Function Documentation	80

7.31.3.1 getDate()	80
7.31.3.2 getDescription()	80
7.31.3.3 getEventName()	80
7.31.3.4 getOrganizerEmail()	80
7.31.3.5 getTitle()	81
7.31.3.6 setDate()	81
7.31.3.7 setDescription()	81
7.31.3.8 setEventName()	81
7.31.3.9 setOrganizerEmail()	81
7.31.3.10 setTitle()	81
7.32 es.deusto.client.windows.Profile Class Reference	82
7.32.1 Detailed Description	82
7.32.2 Constructor & Destructor Documentation	82
7.32.2.1 Profile()	82
7.33 es.deusto.client.windows.SavedEventsWindow Class Reference	82
7.33.1 Detailed Description	83
7.33.2 Constructor & Destructor Documentation	83
7.33.2.1 SavedEventsWindow()	83
7.34 es.deusto.junit.serialization.SerializationTest Class Reference	83
7.34.1 Detailed Description	84
7.34.2 Member Function Documentation	84
7.34.2.1 beforeClass()	84
7.34.2.2 startUp()	84
7.34.2.3 suite()	84
7.34.2.4 testEventInfoFromEvent()	84
7.34.2.5 testEventInfoSetters()	84
7.34.2.6 testOrganizerInfoFromOrganizer()	84
7.34.2.7 testOrganizerInfoSetters()	85
7.34.2.8 testPostInfoFromPost()	85
7.34.2.9 testPostInfoSetters()	85
7.34.2.10 testSignupAttemptOrganizer()	85
7.34.2.11 testSignupAttemptUser()	85
7.34.2.12 testUserInfoFromUser()	85
7.34.2.13 testUserInfoSetters()	85
7.34.2.14 testUserInfoStringConstructor()	86
7.35 es.deusto.server.server.Server Class Reference	86
7.35.1 Detailed Description	86
7.35.2 Constructor & Destructor Documentation	87
7.35.2.1 Server()	87
7.35.3 Member Function Documentation	87
7.35.3.1 attemptNormalLogin()	87
7.35.3.2 attemptNormalSignup()	87

7.35.3.3 attemptNormalUpdate()	88
7.35.3.4 attemptOrganizerDelete()	88
7.35.3.5 attemptOrganizerLogin()	89
7.35.3.6 attemptOrganizerSignup()	89
7.35.3.7 attemptOrganizerUpdate()	90
7.35.3.8 attemptUserDelete()	90
7.35.3.9 createEvent()	91
7.35.3.10 createPost()	91
7.35.3.11 getRecommendedEvents()	92
7.35.3.12 recoverPassword()	92
7.35.3.13 sayHello()	93
7.36 es.deusto.junit.server.ServerTest Class Reference	93
7.36.1 Detailed Description	94
7.36.2 Member Function Documentation	94
7.36.2.1 setUp()	94
7.36.2.2 startUp()	94
7.36.2.3 suite()	94
7.36.2.4 tearDown()	94
7.36.2.5 testCreateEvent()	94
7.36.2.6 testCreatePost()	95
7.36.2.7 testNormalSignup()	95
7.36.2.8 testNormalUpdate()	95
7.36.2.9 testOrganizerLogin()	95
7.36.2.10 testOrganizerSignup()	95
7.36.2.11 testOrganizerUpdate()	95
7.37 es.deusto.serialization.SignupAttempt Class Reference	95
7.37.1 Detailed Description	96
7.37.2 Constructor & Destructor Documentation	96
7.37.2.1 SignupAttempt() [1/2]	96
7.37.2.2 SignupAttempt() [2/2]	96
7.37.3 Member Function Documentation	97
7.37.3.1 buildOrganizer()	97
7.37.3.2 buildUser()	97
7.37.3.3 getCity()	97
7.37.3.4 getCountry()	97
7.37.3.5 getEmail()	97
7.37.3.6 getInterests()	98
7.37.3.7 getName()	98
7.37.3.8 getOrganization()	98
7.37.3.9 getPassword()	98
7.37.3.10 getSavedEvents()	98
7.37.3.11 setCity()	98

7.37.3.12 setCountry()	98
7.37.3.13 setEmail()	99
7.37.3.14 setInterests()	99
7.37.3.15 setName()	99
7.37.3.16 setOrganization()	99
7.37.3.17 setPassword()	99
7.37.3.18 setSaveEvents()	99
7.37.3.19 toString()	99
7.38 es.deusto.server.data.Topic Class Reference	100
7.38.1 Detailed Description	100
7.38.2 Constructor & Destructor Documentation	100
7.38.2.1 Topic() [1/2]	100
7.38.2.2 Topic() [2/2]	100
7.38.3 Member Function Documentation	100
7.38.3.1 getName()	100
7.38.3.2 setName()	101
7.38.3.3 toString()	101
7.39 es.deusto.server.dao.TopicDAO Class Reference	101
7.39.1 Detailed Description	101
7.39.2 Constructor & Destructor Documentation	101
7.39.2.1 TopicDAO()	102
7.39.3 Member Function Documentation	102
7.39.3.1 close()	102
7.39.3.2 deleteTopic()	102
7.39.3.3 getPersistenceManager()	102
7.39.3.4 getTopic()	102
7.39.3.5 storeTopic()	102
7.39.3.6 updateTopic()	103
7.40 es.deusto.serialization.TopicInfo Class Reference	103
7.40.1 Detailed Description	103
7.40.2 Constructor & Destructor Documentation	103
7.40.2.1 TopicInfo() [1/3]	103
7.40.2.2 TopicInfo() [2/3]	104
7.40.2.3 TopicInfo() [3/3]	104
7.40.3 Member Function Documentation	104
7.40.3.1 getName()	104
7.40.3.2 parseTopic()	104
7.40.3.3 setName()	104
7.40.3.4 toString()	105
7.41 es.deusto.server.data.User Class Reference	105
7.41.1 Detailed Description	105
7.41.2 Member Function Documentation	105

7.41.2.1 addEvent()	106
7.41.2.2 addInterest()	106
7.41.2.3 getCity()	106
7.41.2.4 getCountry()	106
7.41.2.5 getInterests()	106
7.41.2.6 getSavedEvents()	106
7.41.2.7 setCity()	106
7.41.2.8 setCountry()	107
7.41.2.9 setInterests()	107
7.41.2.10 setSavedEvents()	107
7.41.2.11 toString()	107
7.42 es.deusto.server.dao.UserDAO Class Reference	107
7.42.1 Detailed Description	108
7.42.2 Constructor & Destructor Documentation	108
7.42.2.1 UserDAO()	108
7.42.3 Member Function Documentation	108
7.42.3.1 close()	108
7.42.3.2 deleteUser()	108
7.42.3.3 getPersistenceManager()	108
7.42.3.4 getUser()	108
7.42.3.5 storeUser()	109
7.42.3.6 updateUser()	109
7.43 es.deusto.client.windows.UserEventsWindow Class Reference	109
7.43.1 Detailed Description	109
7.43.2 Constructor & Destructor Documentation	109
7.43.2.1 UserEventsWindow()	110
7.44 es.deusto.serialization.UserInfo Class Reference	110
7.44.1 Detailed Description	110
7.44.2 Constructor & Destructor Documentation	111
7.44.2.1 UserInfo() [1/3]	111
7.44.2.2 UserInfo() [2/3]	111
7.44.2.3 UserInfo() [3/3]	111
7.44.3 Member Function Documentation	111
7.44.3.1 addInterest()	111
7.44.3.2 getCity()	111
7.44.3.3 getCountry()	111
7.44.3.4 getEmail()	112
7.44.3.5 getInterests()	112
7.44.3.6 getName()	112
7.44.3.7 getSavedEvents()	112
7.44.3.8 setCity()	112
7.44.3.9 setCountry()	112

7.44.3.10 setEmail()	112
7.44.3.11 setInterests()	113
7.44.3.12 setName()	113
7.44.3.13 setSavedEvents()	113

8 File Documentation	115
8.1 src/main/java/es/deusto/client/controller/Controller.java File Reference	115
8.2 src/main/java/es/deusto/client/main/Main.java File Reference	115
8.3 src/main/java/es/deusto/client/windows/ChangePass.java File Reference	115
8.4 src/main/java/es/deusto/client/windows/CodeVerification.java File Reference	116
8.5 src/main/java/es/deusto/client/windows/CreateEvent.java File Reference	116
8.6 src/main/java/es/deusto/client/windows/CreateOrganizerWindow.java File Reference	116
8.7 src/main/java/es/deusto/client/windows/CreateUserWindow.java File Reference	117
8.8 src/main/java/es/deusto/client/windows/EventListItem.java File Reference	117
8.9 src/main/java/es/deusto/client/windows/EventOrganizerWindow.java File Reference	117
8.10 src/main/java/es/deusto/client/windows/EventWindow.java File Reference	117
8.11 src/main/java/es/deusto/client/windows/ForgottenPassword.java File Reference	118
8.12 src/main/java/es/deusto/client/windows/LanguageManager.java File Reference	118
8.13 src/main/java/es/deusto/client/windows/LoginWindow.java File Reference	118
8.14 src/main/java/es/deusto/client/windows/OrganizerEvents.java File Reference	119
8.15 src/main/java/es/deusto/client/windows/OrganizerHome.java File Reference	119
8.16 src/main/java/es/deusto/client/windows/PostCreationWindow.java File Reference	119
8.17 src/main/java/es/deusto/client/windows/Profile.java File Reference	119
8.18 src/main/java/es/deusto/client/windows/SavedEventsWindow.java File Reference	120
8.19 src/main/java/es/deusto/client/windows/UserEventsWindow.java File Reference	120
8.20 src/main/java/es/deusto/serialization/EventInfo.java File Reference	120
8.21 src/main/java/es/deusto/serialization/GenericUserInfo.java File Reference	121
8.22 src/main/java/es/deusto/serialization/LoginAttempt.java File Reference	121
8.23 src/main/java/es/deusto/serialization/OrganizerInfo.java File Reference	121
8.24 src/main/java/es/deusto/serialization/PostInfo.java File Reference	121
8.25 src/main/java/es/deusto/serialization/SignupAttempt.java File Reference	122
8.26 src/main/java/es/deusto/serialization/TopicInfo.java File Reference	122
8.27 src/main/java/es/deusto/serialization/UserInfo.java File Reference	122
8.28 src/main/java/es/deusto/server/dao/DAOFactory.java File Reference	122
8.29 src/main/java/es/deusto/server/dao/EventDAO.java File Reference	123
8.30 src/main/java/es/deusto/server/dao/OrganizerDAO.java File Reference	123
8.31 src/main/java/es/deusto/server/dao/PostDAO.java File Reference	123
8.32 src/main/java/es/deusto/server/dao/TopicDAO.java File Reference	123
8.33 src/main/java/es/deusto/server/dao/UserDAO.java File Reference	124
8.34 src/main/java/es/deusto/server/data/Event.java File Reference	124
8.35 src/main/java/es/deusto/server/data/GenericUser.java File Reference	124
8.36 src/main/java/es/deusto/server/data/Organizer.java File Reference	124

8.37	src/main/java/es/deusto/server/data/Post.java File Reference	125
8.38	src/main/java/es/deusto/server/data/Topic.java File Reference	125
8.39	src/main/java/es/deusto/server/data/User.java File Reference	125
8.40	src/main/java/es/deusto/server/server/AppService.java File Reference	125
8.41	src/main/java/es/deusto/server/server/Server.java File Reference	126
8.42	src/test/java/es/deusto/junit/client/ControllerTest.java File Reference	126
8.43	src/test/java/es/deusto/junit/serialization/SerializationTest.java File Reference	126
8.44	src/test/java/es/deusto/junit/server/ServerTest.java File Reference	126
Index		127

Chapter 1

Deprecated List

Member [es.deusto.client.controller.Controller.attemptNormalLogin](#) (String email, String password, boolean organizer)

beeter to use attemptNormalLogin(email, password) and attemptNormalLoginOrganizer !

Member [es.deusto.server.server.AppService.updateUser](#) ()

no useful for normal use.

Member [es.deusto.server.server.AppService.hello](#) ()

Member [es.deusto.server.server.Server.sayHello](#) ()

Chapter 2

Namespace Index

2.1 Packages

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

es	13
es.deusto	13
es.deusto.client	13
es.deusto.client.controller	13
es.deusto.client.main	14
es.deusto.client.windows	14
es.deusto.junit	15
es.deusto.junit.client	15
es.deusto.junit.serialization	15
es.deusto.junit.server	15
es.deusto.serialization	15
es.deusto.server	16
es.deusto.server.dao	16
es.deusto.server.data	16
es.deusto.server.server	17

Chapter 3

Hierarchical Index

3.1 Class Hierarchy

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

es.deusto.server.server.AppService	19
es.deusto.client.controller.Controller	28
es.deusto.server.dao.DAOFactory	42
es.deusto.server.data.Event	44
es.deusto.server.dao.EventDAO	48
es.deusto.serialization.EventInfo	51
es.deusto.server.data.GenericUser	58
es.deusto.server.data.Organizer	67
es.deusto.server.data.User	105
es.deusto.serialization.GenericUserInfo	60
es.deusto.serialization.OrganizerInfo	72
es.deusto.serialization.UserInfo	110
es.deusto.client.windows.LanguageManager	61
es.deusto.serialization.LoginAttempt	62
es.deusto.client.main.Main	66
es.deusto.server.dao.OrganizerDAO	69
es.deusto.server.data.Post	74
es.deusto.server.dao.PostDAO	77
es.deusto.serialization.PostInfo	79
es.deusto.junit.serialization.SerializationTest	83
es.deusto.server.server.Server	86
es.deusto.junit.server.ServerTest	93
es.deusto.serialization.SignupAttempt	95
es.deusto.server.data.Topic	100
es.deusto.server.dao.TopicDAO	101
es.deusto.serialization.TopicInfo	103
es.deusto.server.dao.UserDAO	107
ActionListener	
es.deusto.client.windows.CreateEvent	39
es.deusto.client.windows.CreateOrganizerWindow	40
es.deusto.client.windows.CreateUserWindow	41
JerseyTest	
es.deusto.junit.client.ControllerTest	36
JFrame	
es.deusto.client.windows.ChangePass	26

es.deusto.client.windows.CodeVerification	27
es.deusto.client.windows.CreateEvent	39
es.deusto.client.windows.CreateOrganizerWindow	40
es.deusto.client.windows.CreateUserWindow	41
es.deusto.client.windows.EventOrganizerWindow	56
es.deusto.client.windows.EventWindow	57
es.deusto.client.windows.ForgottenPassword	58
es.deusto.client.windows.LogInWindow	65
es.deusto.client.windows.OrganizerEvents	70
es.deusto.client.windows.OrganizerHome	71
es.deusto.client.windows.PostCreationWindow	77
es.deusto.client.windows.Profile	82
es.deusto.client.windows.SavedEventsWindow	82
es.deusto.client.windows.UserEventsWindow	109
JPanel	
es.deusto.client.windows.EventListItem	55

Chapter 4

Class Index

4.1 Class List

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

es.deusto.server.server.AppService	19
es.deusto.client.windows.ChangePass	
This class is the one used to create a window to change the user password	26
es.deusto.client.windows.CodeVerification	27
es.deusto.client.controller.Controller	
This class is the one carrying out all the communication with the server	28
es.deusto.junit.client.ControllerTest	
This classes is the one that tests the controller component to see if it is working as intended	36
es.deusto.client.windows.CreateEvent	
This class is the one used to create a window that it allows the user to create a new event	39
es.deusto.client.windows.CreateOrganizerWindow	
This class is the one used to create a window that allows to create a new organizer	40
es.deusto.client.windows.CreateUserWindow	
This class is the one used to create a window that allows to create a new user	41
es.deusto.server.dao.DAOFactory	
This class is used to create a DAO factory that is used by all of the other DAO classes, to create them or to close them	42
es.deusto.server.data.Event	
This classes is the one that defines an event and it is used to create one	44
es.deusto.server.dao.EventDAO	
This classes is the one used to create an event DAO	48
es.deusto.serialization.EventInfo	
DTO equivalent class of the Event class from the server	51
es.deusto.client.windows.EventListItem	
This class is the one used to create a window that allows the user to have an event list	55
es.deusto.client.windows.EventOrganizerWindow	
This class is the one used to create a window that allows to an organizer to change or administrate one of his events	56
es.deusto.client.windows.EventWindow	
This class is the one used to create a window that allows to display all the information of one event	57
es.deusto.client.windows.ForgottenPassword	
This class is the one used to create a window that allows a user to log in when he forgets the password	58
es.deusto.server.data.GenericUser	
This classes is the father class for normal user and organizer	58

es.deusto.serialization.GenericUserInfo	
This class works as a Tag for the Classes related to the User and Organizer	60
es.deusto.client.windows.LanguageManager	
This class is used to manage the different languages supported by the application	61
es.deusto.serialization.LoginAttempt	
This class encapsulates all the information required when a user (Either normal User or Organizer) logs in	62
es.deusto.client.windows.LoginWindow	
This class is the one used to create a window that allows a user/organizer to log in to the application	65
es.deusto.client.main.Main	
This class is the one that starts the application	66
es.deusto.server.data.Organizer	
This classes is the one that defines an organizer and it is used to create one	67
es.deusto.server.dao.OrganizerDAO	
This classes is the one used to create an organizer DAO	69
es.deusto.client.windows.OrganizerEvents	
This class is the one used to create a window that allows an organizer to have all of his events	70
es.deusto.client.windows.OrganizerHome	
This class is the one used to create a window that allows an organizer to have a home window where he can access to all of the functionalities of the application	71
es.deusto.serialization.OrganizerInfo	
DTO equivalent class of the Organizer class from the server	72
es.deusto.server.data.Post	
This classes is the one that defines a post and it is used to create one	74
es.deusto.client.windows.PostCreationWindow	
This class is the one used to create a window that allows an organizer to have like a double check of the event he has just created	77
es.deusto.server.dao.PostDAO	
This classes is the one used to create a DAO for a post	77
es.deusto.serialization.PostInfo	
DTO equivalent class of the Post class from the server	79
es.deusto.client.windows.Profile	
This class is the one used to create a window that allows a user/organizer to have a profile with all of his information	82
es.deusto.client.windows.SavedEventsWindow	
This class is the one used to create a window that allows to scroll, search and choose from all of the events stored in the database	82
es.deusto.junit.serialization.SerializationTest	
This classes is the one that tests funcionalities such as creating an event, creating a user, orginzer, crea ting posts and topics	83
es.deusto.server.server.Server	
This class is the one that receives the requests from the client	86
es.deusto.junit.server.ServerTest	
This classes is used to create several tests to see if the server is working as intended	93
es.deusto.serialization.SignupAttempt	
This class encapsulates all the information required when a user (Either normal User or Organizer) signs up	95
es.deusto.server.data.Topic	
This classes is the one that defines a topic and it is used to create one	100
es.deusto.server.dao.TopicDAO	
This classes is the one used to create a DAO a topic	101
es.deusto.serialization.TopicInfo	
DTO equivalent class of the Topic class from the server	103
es.deusto.server.data.User	
This classes is the one that defines a normal user and it is used to create one	105
es.deusto.server.dao.UserDAO	
This classes is the one used to create an user DAO	107

[es.deusto.client.windows.UserEventsWindow](#)

This class is the one used to create a window that allows a user to scroll, search and select from a list of events he previously has chosen from 109

[es.deusto.serialization.UserInfo](#)

DTO equivalent class of the User class from the server 110

Chapter 5

File Index

5.1 File List

Here is a list of all files with brief descriptions:

src/main/java/es/deusto/client/controller/Controller.java	115
src/main/java/es/deusto/client/main/Main.java	115
src/main/java/es/deusto/client/windows/ChangePass.java	115
src/main/java/es/deusto/client/windows/CodeVerification.java	116
src/main/java/es/deusto/client/windows/CreateEvent.java	116
src/main/java/es/deusto/client/windows/CreateOrganizerWindow.java	116
src/main/java/es/deusto/client/windows/CreateUserWindow.java	117
src/main/java/es/deusto/client/windows/EventListItem.java	117
src/main/java/es/deusto/client/windows/EventOrganizerWindow.java	117
src/main/java/es/deusto/client/windows/EventWindow.java	117
src/main/java/es/deusto/client/windows/ForgottenPassword.java	118
src/main/java/es/deusto/client/windows/LanguageManager.java	118
src/main/java/es/deusto/client/windows/LoginWindow.java	118
src/main/java/es/deusto/client/windows/OrganizerEvents.java	119
src/main/java/es/deusto/client/windows/OrganizerHome.java	119
src/main/java/es/deusto/client/windows/PostCreationWindow.java	119
src/main/java/es/deusto/client/windows/Profile.java	119
src/main/java/es/deusto/client/windows/SavedEventsWindow.java	120
src/main/java/es/deusto/client/windows/UserEventsWindow.java	120
src/main/java/es/deusto/serialization/EventInfo.java	120
src/main/java/es/deusto/serialization/GenericUserInfo.java	121
src/main/java/es/deusto/serialization/LoginAttempt.java	121
src/main/java/es/deusto/serialization/OrganizerInfo.java	121
src/main/java/es/deusto/serialization/PostInfo.java	121
src/main/java/es/deusto/serialization/SignupAttempt.java	122
src/main/java/es/deusto/serialization/TopicInfo.java	122
src/main/java/es/deusto/serialization/UserInfo.java	122
src/main/java/es/deusto/server/dao/DAOFactory.java	122
src/main/java/es/deusto/server/dao/EventDAO.java	123
src/main/java/es/deusto/server/dao/OrganizerDAO.java	123
src/main/java/es/deusto/server/dao/PostDAO.java	123
src/main/java/es/deusto/server/dao/TopicDAO.java	123
src/main/java/es/deusto/server/dao/UserDAO.java	124
src/main/java/es/deusto/server/data/Event.java	124
src/main/java/es/deusto/server/data/GenericUser.java	124

src/main/java/es/deusto/server/data/ Organizer.java	124
src/main/java/es/deusto/server/data/ Post.java	125
src/main/java/es/deusto/server/data/ Topic.java	125
src/main/java/es/deusto/server/data/ User.java	125
src/main/java/es/deusto/server/server/ AppService.java	125
src/main/java/es/deusto/server/server/ Server.java	126
src/test/java/es/deusto/junit/client/ ControllerTest.java	126
src/test/java/es/deusto/junit/serialization/ SerializationTest.java	126
src/test/java/es/deusto/junit/server/ ServerTest.java	126

Chapter 6

Namespace Documentation

6.1 Package es

Packages

- package [deusto](#)

6.2 Package es.deusto

Packages

- package [client](#)
- package [junit](#)
- package [serialization](#)
- package [server](#)

6.3 Package es.deusto.client

Packages

- package [controller](#)
- package [main](#)
- package [windows](#)

6.4 Package es.deusto.client.controller

Classes

- class [Controller](#)

This class is the one carrying out all the communication with the server.

6.5 Package es.deusto.client.main

Classes

- class [Main](#)

This class is the one that starts the application.

6.6 Package es.deusto.client.windows

Classes

- class [ChangePass](#)

This class is the one used to create a window to change the user password.

- class [CodeVerification](#)

- class [CreateEvent](#)

This class is the one used to create a window that it allows the user to create a new event.

- class [CreateOrganizerWindow](#)

This class is the one used to create a window that allows to create a new organizer.

- class [CreateUserWindow](#)

This class is the one used to create a window that allows to create a new user.

- class [EventListItem](#)

This class is the one used to create a window that allows the user to have an event list.

- class [EventOrganizerWindow](#)

This class is the one used to create a window that allows to an organizer to change or administrate one of his events.

- class [EventWindow](#)

This class is the one used to create a window that allows to display all the information of one event.

- class [ForgottenPassword](#)

This class is the one used to create a window that allows a user to log in when he forgets the password.

- class [LanguageManager](#)

This class is used to manage the different languages supported by the application.

- class [LoginWindow](#)

This class is the one used to create a window that allows a user/organizer to log in to the application.

- class [OrganizerEvents](#)

This class is the one used to create a window that allows an organizer to have all of his events.

- class [OrganizerHome](#)

This class is the one used to create a window that allows an organizer to have a home window where he can access to all of the functionalities of the application.

- class [PostCreationWindow](#)

This class is the one used to create a window that allows an organizer to have like a double check of the event he has just created.

- class [Profile](#)

This class is the one used to create a window that allows a user/organizer to have a profile with all of his information.

- class [SavedEventsWindow](#)

This class is the one used to create a window that allows to scroll, search and choose from all of the events stored in the database.

- class [UserEventsWindow](#)

This class is the one used to create a window that allows a user to scroll, search and select from a list of events he previously has chosen from.

6.7 Package es.deusto.junit

Packages

- package [client](#)
- package [serialization](#)
- package [server](#)

6.8 Package es.deusto.junit.client

Classes

- class [ControllerTest](#)

This classes is the one that tests the controller component to see if it is working as intended.

6.9 Package es.deusto.junit.serialization

Classes

- class [SerializationTest](#)

This classes is the one that tests funcionalities such as creating an event, creating a user, orginzer, crea ting posts and topics.

6.10 Package es.deusto.junit.server

Classes

- class [ServerTest](#)

This classes is used to create several tests to see if the server is working as intended.

6.11 Package es.deusto.serialization

Classes

- class [EventInfo](#)

DTO equivalent class of the Event class from the server.

- class [GenericUserInfo](#)

This class works as a Tag for the Classes related to the User and Organizer.

- class [LoginAttempt](#)

This class encapsulates all the information required when a user (Either normal User or Organizer) logs in.

- class [OrganizerInfo](#)

DTO equivalent class of the Organizer class from the server.

- class [PostInfo](#)

DTO equivalent class of the Post class from the server.

- class [SignupAttempt](#)

This class encapsulates all the information required when a user (Either normal User or Organizer) signs up.

- class [TopicInfo](#)

DTO equivalent class of the Topic class from the server.

- class [UserInfo](#)

DTO equivalent class of the User class from the server.

6.11.1 Detailed Description

All the ...Info classes work as a middleman for the client and server bussines objects. Due to the functionality of this classes their structure is similar to the 'original' ones in the server Examples: [UserInfo](#): User [EventInfo](#): Event Also the ...Info clases are the ones shown in the GUI.

6.12 Package es.deusto.server

Packages

- package [dao](#)
- package [data](#)
- package [server](#)

6.13 Package es.deusto.server.dao

Classes

- class [DAOFactory](#)
This class is used to create a DAO factory that is used by all of the other DAO classes, to create them or to close them.
- class [EventDAO](#)
This classes is the one used to create an event DAO.
- class [OrganizerDAO](#)
This classes is the one used to create an organizer DAO.
- class [PostDAO](#)
This classes is the one used to create a DAO for a post.
- class [TopicDAO](#)
This classes is the one used to create a DAO a topic.
- class [UserDAO](#)
This classes is the one used to create an user DAO.

6.14 Package es.deusto.server.data

Classes

- class [Event](#)
This classes is the one that defines an event and it is used to create one.
- class [GenericUser](#)
This classes is the father class for normal user and organizer.
- class [Organizer](#)
This classes is the one that defines an organizer and it is used to create one.
- class [Post](#)
This classes is the one that defines a post and it is used to create one.
- class [Topic](#)
This classes is the one that defines a topic and it is used to create one.
- class [User](#)
This classes is the one that defines a normal user and it is used to create one.

6.15 Package es.deusto.server.server

Classes

- class [AppService](#)
- class [Server](#)

This class is the one that receives the requests from the client.

Chapter 7

Class Documentation

7.1 es.deusto.server.server.AppService Class Reference

Public Member Functions

- [AppService](#) ()
This class will include methods representing the different functionalities of the server.
- [User attemptNormalLogin](#) ([LoginAttempt](#) login)
This method is used for the login process of regular users.
- [Organizer attemptOrganizerLogin](#) ([LoginAttempt](#) login)
This method is used for the login process of organizers.
- [User attemptNormalSignup](#) ([SignupAttempt](#) signup)
This method is used for the signup process of regular users.
- [Organizer attemptOrganizerSignup](#) ([SignupAttempt](#) signup)
This method is used for the signup process of organizers.
- [User attemptNormalUpdate](#) ([SignupAttempt](#) signup)
This method is used for the update process of regular users (not organizers).
- [Organizer attemptOrganizerUpdate](#) ([SignupAttempt](#) signup)
This method is used for the updating process of organizers.
- boolean [deleteUser](#) (String email)
This method is used for the delete process of users.
- boolean [deleteOrganizer](#) (String email)
This method is used for the delete process of organizers.
- [Event createEvent](#) ([EventInfo](#) eventInfo)
Method used for storing new Events in the database, used only by the organizers.
- ArrayList< [EventInfo](#) > [getRecommendedEvents](#) ([SignupAttempt](#) signupAttempt)
Creates a list of events that coincide with the interests (Topic) of a user.
- void [recoverPassword](#) ([LoginAttempt](#) login)
This method is used for new password generation.
- [Post createPost](#) ([PostInfo](#) info)
This method is used for the creation of new Post from a given Event.
- void [updateUser](#) ()
Used for testing, creates an instance of a user and updates it in the database.
- void [hello](#) ()
Used for testing.

7.1.1 Constructor & Destructor Documentation

7.1.1.1 AppService()

```
es.deusto.server.server.AppService.AppService ( )
```

This class will include methods representing the different functionalities of the server.

This class should be called by the [Server](#) class, so that the [Server](#) is abstracted from the business logic of the project. In order to access to the database, use the DAO objects (create them using DAOFactory class).

7.1.2 Member Function Documentation

7.1.2.1 attemptNormalLogin()

```
User es.deusto.server.server.AppService.attemptNormalLogin (
    LoginAttempt login )
```

This method is used for the login process of regular users.

Receives a LoginAttempt object with the email and password and compares it with the copy stored in the database.

Parameters

<i>login</i>	LoginAttempt with the requester's email and password.
--------------	---

Returns

User object to which the login email corresponds. If the credentials are invalid, null is returned.

Since

Sprint 1

7.1.2.2 attemptNormalSignup()

```
User es.deusto.server.server.AppService.attemptNormalSignup (
    SignupAttempt signup )
```

This method is used for the signup process of regular users.

Receives a SignupAttempt object with the information of the user and tries to submit it to the database.

Parameters

<i>signup</i>	SignupAttempt with the user's signup data (name, email, password and city).
---------------	---

Returns

User object that has been created as a result of the request. If the email is already in use, null is returned.

Since

Sprint 1

7.1.2.3 attemptNormalUpdate()

```
User es.deusto.server.server.AppService.attemptNormalUpdate (
    SignupAttempt signup )
```

This method is used for the update process of regular users (not organizers).

Receives a SignupAttempt object and tries to update an existing user in the database.

Parameters

<i>signup</i>	SignupAttempt with the user's data (name, email, an empty password, city).
---------------	--

Since

Sprint 2

Returns

User object that has been created as a result of the request. If the email doesn't exist null is returned

7.1.2.4 attemptOrganizerLogin()

```
Organizer es.deusto.server.server.AppService.attemptOrganizerLogin (
    LoginAttempt login )
```

This method is used for the login process of organizers.

Receives a LoginAttempt object with the email and password and compares it with the copy stored in the database.

Parameters

<i>login</i>	LoginAttempt with the requester's email and password.
--------------	---

Returns

Organizer object to which the login email corresponds. If the credentials are invalid, null is returned.

Since

Sprint 2

7.1.2.5 attemptOrganizerSignup()

```
Organizer es.deusto.server.server.AppService.attemptOrganizerSignup (
    SignupAttempt signup )
```

This method is used for the signup process of organizers.

Receives a SignupAttempt object with the information of the user and tries to submit it to the database.

Parameters

<i>signup</i>	SignupAttempt with the organizer's signup data - name, email, password and organization
---------------	---

Returns

Organizer object that has been created as a result of the request. If the email is already in use, null is returned.

Since

Sprint 2

7.1.2.6 attemptOrganizerUpdate()

```
Organizer es.deusto.server.server.AppService.attemptOrganizerUpdate (
    SignupAttempt signup )
```

This method is used for the updating process of organizers.

Receives a SignupAttempt object and tries to update an existing organizer in the database.

Parameters

<i>signup</i>	SignupAttempt object with the requester's data (name, email, empty password, organization).
---------------	---

Since

Sprint 2

Returns

Organizer object that has been updated as a result of the request. If the email doesn't exist null is returned.

7.1.2.7 createEvent()

```
Event es.deusto.server.server.AppService.createEvent (
    EventInfo eventInfo )
```

Method used for storing new Events in the database, used only by the organizers.

Receives an EventInfo, transforms it into a Event and stores it in the database.

Parameters

<i>eventInfo</i>	
------------------	--

Returns

Event object of the newly stored event.

Since

sprint 2

7.1.2.8 createPost()

```
Post es.deusto.server.server.AppService.createPost (
    PostInfo info )
```

This method is used for the creation of new Post from a given Event.

It is invoked whenever a POST request is made to the following path: /createEvent.

Parameters

<i>info</i>	PostInfo object with the data of the event that will be created.
-------------	--

Returns

Post object containing the information of the stored result.

Since

Sprint 2

7.1.2.9 deleteOrganizer()

```
boolean es.deusto.server.server.AppService.deleteOrganizer (  
    String email )
```

This method is used for the delete process of organizers.

Receives an String with the email and tries to delete the organizer account related to it from the database.

Parameters

<i>email</i>	of the organizer account to be deleted.
--------------	---

Since

Sprint 2

Returns

boolean True if the email belongs to a organizer and the delete is complete successfully, false otherwise.

7.1.2.10 deleteUser()

```
boolean es.deusto.server.server.AppService.deleteUser (  
    String email )
```

This method is used for the delete process of users.

Receives an String with the email and tries to delete the account related to it from the database.

Parameters

<i>email</i>	of the account to be deleted.
--------------	-------------------------------

Since

Sprint 2

Returns

boolean True if the email belongs to a user and the delete is complete successfully, false otherwise.

7.1.2.11 getRecommendedEvents()

```
ArrayList<EventInfo> es.deusto.server.server.AppService.getRecommendedEvents (
    SignupAttempt signupAttempt )
```

Creates a list of events that coincide with the interests (Topic) of a user.

This method reads the interests of a User, and returns the adequate Events from the database. It is used for making automatic recommendations, a user should be able to see a list of events that he/she should be interested.

Parameters

<i>signupAttempt</i>	SignupAttempt object containing the interests of the user.
----------------------	--

Returns

ArrayList of EventInfo list of events that coincide with the interests of a user.

Since

Sprint 3

7.1.2.12 hello()

```
void es.deusto.server.server.AppService.hello ( )
```

Used for testing.

Deprecated

7.1.2.13 recoverPassword()

```
void es.deusto.server.server.AppService.recoverPassword (
    LoginAttempt login )
```

This method is used for new password generation.

It receives a LoginAttempt with the email of the user that wants to recover his/her password, generates a new password for that user and updates the data in the database. It also sends an email to the received email with the new password so that the user can log in. If there is no account in the database with that same email linked to it, the method does nothing.

Parameters

<i>login</i>	LoginAttempt with the user's email
--------------	------------------------------------

Since

Sprint 1

7.1.2.14 updateUser()

```
void es.deusto.server.server.AppService.updateUser ( )
```

Used for testing, creates an instance of a user and updates it in the database.

Deprecated no useful for normal use.

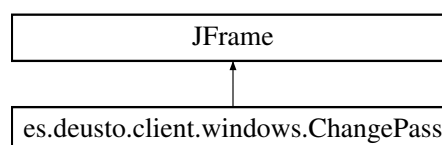
The documentation for this class was generated from the following file:

- [src/main/java/es/deusto/server/server/AppService.java](#)

7.2 es.deusto.client.windows.ChangePass Class Reference

This class is the one used to create a window to change the user password.

Inheritance diagram for es.deusto.client.windows.ChangePass:



Public Member Functions

- [ChangePass](#) ([Controller](#) controller)
Create the frame.

7.2.1 Detailed Description

This class is the one used to create a window to change the user password.

7.2.2 Constructor & Destructor Documentation

7.2.2.1 ChangePass()

```
es.deusto.client.windows.ChangePass.ChangePass (
    Controller controller )
```

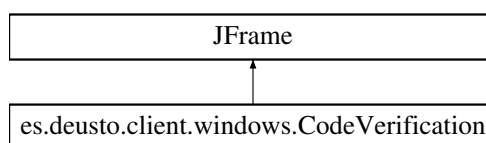
Create the frame.

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

- src/main/java/es/deusto/client/windows/[ChangePass.java](#)

7.3 es.deusto.client.windows.CodeVerification Class Reference

Inheritance diagram for es.deusto.client.windows.CodeVerification:



Public Member Functions

- [CodeVerification](#) (String code, [Controller](#) controller)
Create the frame.

7.3.1 Constructor & Destructor Documentation

7.3.1.1 CodeVerification()

```
es.deusto.client.windows.CodeVerification.CodeVerification (
    String code,
    Controller controller )
```

Create the frame.

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

- [src/main/java/es/deusto/client/windows/CodeVerification.java](#)

7.4 es.deusto.client.controller.Controller Class Reference

This class is the one carrying out all the communication with the server.

Public Member Functions

- [Controller](#) (String hostname, String port)
Constructor.
- [Controller](#) (WebTarget target)
Constructor.
- [UserInfo](#) [getUser](#) ()
- [OrganizerInfo](#) [getOrganize](#) ()
- void [setOrganizer](#) ([OrganizerInfo](#) org)
- void [setUser](#) ([UserInfo](#) u)
- boolean [attemptNormalLogin](#) (String email, String password)
Invokes a normal login attempt call to the server.
- boolean [attemptNormalLoginOrganizer](#) (String email, String password)
Invokes an organizer login attempt call to the server.
- boolean [attemptNormalLogin](#) (String email, String password, boolean organizer)
Invokes a normal login attempt call to the server.
- boolean [attemptNormalSignup](#) (String email, String password, String name, String city, String country, Array↔
List< [TopicInfo](#) >interests)
Invokes a normal signup attempt call to the server.
- boolean [attemptOrganizerSignup](#) (String email, String password, String name, String organization)
Invokes an organizer signup attempt call to the server.
- boolean [attemptNormalUpdate](#) ()
Invokes a normal user update call to the server.
- boolean [attemptOrganizerUpdate](#) ()
Invokes an organizer update call to the server.
- boolean [attemptUserDelete](#) ()
Invokes a user delete call to the server.
- boolean [attemptOrganizerDelete](#) ()
Invokes an organizer delete call to the server.
- boolean [createEvent](#) ([EventInfo](#) eventInfo)
Invokes an event creation call to the server.
- boolean [createPost](#) ([EventInfo](#) eventInfo, [PostInfo](#) postInfo)
Invokes a post creation call to the server.
- [LanguageManager](#) [getLanguageManager](#) ()
- ArrayList< [EventInfo](#) > [getRecommendations](#) ()
Invokes a recommendations request call to the server.

7.4.1 Detailed Description

This class is the one carrying out all the communication with the server.

The GUI components use it as a middle-man in order to retrieve or update information of the server.

This class will contain the UserInfo property containing the information of the user who has logged in, and will offer it to the components of the GUI.

7.4.2 Constructor & Destructor Documentation

7.4.2.1 Controller() [1/2]

```
es.deusto.client.controller.Controller.Controller (
    String hostname,
    String port )
```

Constructor.

Parameters

<i>hostname</i>	A String object with the path of the server
<i>port</i>	A String object with the port number of the server

7.4.2.2 Controller() [2/2]

```
es.deusto.client.controller.Controller.Controller (
    WebTarget target )
```

Constructor.

Parameters

<i>target</i>	WebTarget instance pointing to the server (used for testing)
---------------	--

Since

Sprint 2

7.4.3 Member Function Documentation

7.4.3.1 attemptNormalLogin() [1/2]

```
boolean es.deusto.client.controller.Controller.attemptNormalLogin (
    String email,
    String password )
```

Invokes a normal login attempt call to the server.

This method is invoked by the login button in the GUI, and it makes a login request to the server with the specified email and password. It is used only for regular users, not organizers.

Parameters

<i>email</i>	Email of the user taken from the GUI
<i>password</i>	Password taken from the GUI

Returns

true if the login was successful, otherwise it returns false

Since

Sprint 1

7.4.3.2 attemptNormalLogin() [2/2]

```
boolean es.deusto.client.controller.Controller.attemptNormalLogin (
    String email,
    String password,
    boolean organizer )
```

Invokes a normal login attempt call to the server.

This method is invoked by the login button in the GUI, and it makes a login request to the server with the specified email and password. The class of object must be indicated in the boolean input. This method works as a solution for both organizers and logins

Parameters

<i>email</i>	email of the User or Organizer
<i>password</i>	password of the of the UserInfo or OrganizerInfo
<i>organizer</i>	boolean that indicates whether it's a User or Organizer

Deprecated beeter to use attemptNormalLogin(email, password) and attemptNormalLoginOrganizer !

Since

Sprint 2

Returns

boolean if succesful

7.4.3.3 attemptNormalLoginOrganizer()

```
boolean es.deusto.client.controller.Controller.attemptNormalLoginOrganizer (
    String email,
    String password )
```

Invokes an organizer login attempt call to the server.

This method is invoked by the login button in the GUI, and it makes a login request to the server with the specified email and password. It is used only for organizers.

Parameters

<i>email</i>	email of the Organizer
<i>password</i>	password of the organizer

Returns

true if succesful

Since

Sprint 2

7.4.3.4 attemptNormalSignup()

```
boolean es.deusto.client.controller.Controller.attemptNormalSignup (
    String email,
    String password,
    String name,
    String city,
    String country,
    ArrayList< TopicInfo > interests )
```

Invokes a normal signup attempt call to the server.

This method is invoked by the signup button in the GUI, and makes a signup request to the server with the specified fields (email, password, name and city). It is used only for regular users, not organizers.

Parameters

<i>email</i>	Email String taken from the GUI
<i>password</i>	Password String taken from the GUI
<i>name</i>	Name String taken from the GUI
<i>city</i>	City String taken from the GUI
<i>country</i>	Country String taken from the GUI
<i>interests</i>	List of TopicInfo

Returns

true if the signup process was successful, otherwise it returns false

Since

Sprint 1

7.4.3.5 attemptNormalUpdate()

```
boolean es.deusto.client.controller.Controller.attemptNormalUpdate ( )
```

Invokes a normal user update call to the server.

Updates the user stored in the [Controller](#), in order to use this function a user must log in before.

Returns

true if the process is successful

Since

Sprint 2

7.4.3.6 attemptOrganizerDelete()

```
boolean es.deusto.client.controller.Controller.attemptOrganizerDelete ( )
```

Invokes an organizer delete call to the server.

Deletes the organizer stored in the [Controller](#).

Returns

true if the process is successful, false if not

Since

Sprint 2

Returns

7.4.3.7 attemptOrganizerSignup()

```
boolean es.deusto.client.controller.Controller.attemptOrganizerSignup (
    String email,
    String password,
    String name,
    String organization )
```

Invokes an organizer signup attempt call to the server.

This method is invoked by the signup button in the GUI, and makes a signup request to the server with the specified fields (email, password, name and organization). It is used only for organizers.

Parameters

<i>email</i>	Email String taken from the GUI
<i>password</i>	Password String taken from the GUI
<i>name</i>	Name String taken from the GUI
<i>organization</i>	String

Since

Sprint 2

Returns

true if the signup process was successful, otherwise it returns false

7.4.3.8 attemptOrganizerUpdate()

```
boolean es.deusto.client.controller.Controller.attemptOrganizerUpdate ( )
```

Invokes an organizer update call to the server.

Updates the Organizer stored in the [Controller](#), in order to use this function a Organizer must log in before.

Returns

true if the process is successful

Since

Sprint 2

7.4.3.9 attemptUserDelete()

```
boolean es.deusto.client.controller.Controller.attemptUserDelete ( )
```

Invokes a user delete call to the server.

Deletes the user stored in the [Controller](#).

Returns

true if the process is successful, false if not

Since

Sprint 2

7.4.3.10 createEvent()

```
boolean es.deusto.client.controller.Controller.createEvent (
    EventInfo eventInfo )
```

Invokes an event creation call to the server.

Sends EventInfo to the server, to use this method the organizer must be logged in.

Parameters

<i>eventInfo</i>	EventInfo to be sent to the server
------------------	------------------------------------

Returns

true if succesful, false otherwise

Since

sprint 2

7.4.3.11 createPost()

```
boolean es.deusto.client.controller.Controller.createPost (
    EventInfo eventInfo,
    PostInfo postInfo )
```

Invokes a post creation call to the server.

Adds a post on the indicated Event and notifies the server.

Parameters

<i>eventInfo</i>	EventInfo where the post is going to be held.
<i>postInfo</i>	PostInfo to be added.

Returns

true if succesful

Since

sprint 2

7.4.3.12 getLanguageManager()

`LanguageManager` es.deusto.client.controller.Controller.getLanguageManager ()

7.4.3.13 getOrganize()

`OrganizerInfo` es.deusto.client.controller.Controller.getOrganize ()

7.4.3.14 getRecommendations()

`ArrayList<EventInfo>` es.deusto.client.controller.Controller.getRecommendations ()

Invokes a recommendations request call to the server.

Sends to the server the information of the logged User in order to get event recomendations

Returns

a list of recommended events

Since

Sprint 3

7.4.3.15 getUser()

```
UserInfo es.deusto.client.controller.Controller.getUser ( )
```

7.4.3.16 setOrganizer()

```
void es.deusto.client.controller.Controller.setOrganizer (
    OrganizerInfo org )
```

7.4.3.17 setUser()

```
void es.deusto.client.controller.Controller.setUser (
    UserInfo u )
```

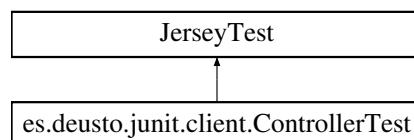
The documentation for this class was generated from the following file:

- [src/main/java/es/deusto/client/controller/Controller.java](#)

7.5 es.deusto.junit.client.ControllerTest Class Reference

This classes is the one that tests the controller component to see if it is working as intended.

Inheritance diagram for es.deusto.junit.client.ControllerTest:



Public Member Functions

- Application [configure](#) ()
- void [setUpChild](#) ()
- void [testNormalLogin](#) ()
- void [testOrganizerLogin](#) ()
- void [testNormalSignupAndDelete](#) ()
- void [testOrganizerSignupAndDelete](#) ()
- void [testUserUpdate](#) ()
- void [tearDownChild](#) ()

Static Public Member Functions

- static void `startUp` ()
- static junit.framework.Test `suite` ()

Public Attributes

- ContiPerfRule `rule` = new ContiPerfRule()

7.5.1 Detailed Description

This classes is the one that tests the controller component to see if it is working as intended.

7.5.2 Member Function Documentation

7.5.2.1 `configure()`

```
Application es.deusto.junit.client.ControllerTest.configure ( )
```

7.5.2.2 `setUpChild()`

```
void es.deusto.junit.client.ControllerTest.setUpChild ( )
```

7.5.2.3 `startUp()`

```
static void es.deusto.junit.client.ControllerTest.startUp ( ) [static]
```

7.5.2.4 `suite()`

```
static junit.framework.Test es.deusto.junit.client.ControllerTest.suite ( ) [static]
```

7.5.2.5 tearDownChild()

```
void es.deusto.junit.client.ControllerTest.tearDownChild ( )
```

7.5.2.6 testNormalLogin()

```
void es.deusto.junit.client.ControllerTest.testNormalLogin ( )
```

7.5.2.7 testNormalSignupAndDelete()

```
void es.deusto.junit.client.ControllerTest.testNormalSignupAndDelete ( )
```

7.5.2.8 testOrganizerLogin()

```
void es.deusto.junit.client.ControllerTest.testOrganizerLogin ( )
```

7.5.2.9 testOrganizerSignupAndDelete()

```
void es.deusto.junit.client.ControllerTest.testOrganizerSignupAndDelete ( )
```

7.5.2.10 testUserUpdate()

```
void es.deusto.junit.client.ControllerTest.testUserUpdate ( )
```

7.5.3 Member Data Documentation

7.5.3.1 rule

```
ContiPerfRule es.deusto.junit.client.ControllerTest.rule = new ContiPerfRule()
```

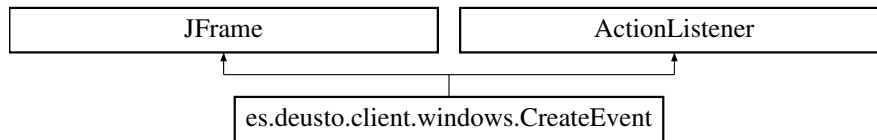
The documentation for this class was generated from the following file:

- [src/test/java/es/deusto/junit/client/ControllerTest.java](#)

7.6 es.deusto.client.windows.CreateEvent Class Reference

This class is the one used to create a window that it allows the user to create a new event.

Inheritance diagram for es.deusto.client.windows.CreateEvent:



Public Member Functions

- [CreateEvent](#) ([Controller](#) controller)
- void [actionPerformed](#) ([ActionEvent](#) e)

7.6.1 Detailed Description

This class is the one used to create a window that it allows the user to create a new event.

7.6.2 Constructor & Destructor Documentation

7.6.2.1 CreateEvent()

```
es.deusto.client.windows.CreateEvent.CreateEvent (
    Controller controller )
```

7.6.3 Member Function Documentation

7.6.3.1 actionPerformed()

```
void es.deusto.client.windows.CreateEvent.actionPerformed (
    ActionEvent e )
```

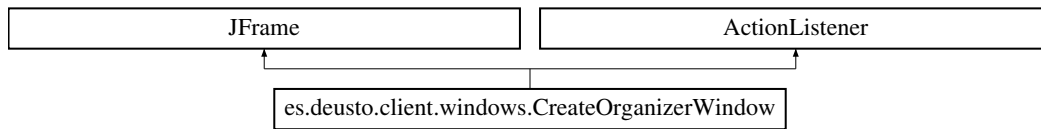
The documentation for this class was generated from the following file:

- [src/main/java/es/deusto/client/windows/CreateEvent.java](#)

7.7 es.deusto.client.windows.CreateOrganizerWindow Class Reference

This class is the one used to create a window that allows to create a new organizer.

Inheritance diagram for es.deusto.client.windows.CreateOrganizerWindow:



Public Member Functions

- [CreateOrganizerWindow](#) ([Controller](#) controller)
- void [actionPerformed](#) ([ActionEvent](#) e)

7.7.1 Detailed Description

This class is the one used to create a window that allows to create a new organizer.

7.7.2 Constructor & Destructor Documentation

7.7.2.1 CreateOrganizerWindow()

```
es.deusto.client.windows.CreateOrganizerWindow.CreateOrganizerWindow (
    Controller controller )
```

7.7.3 Member Function Documentation

7.7.3.1 actionPerformed()

```
void es.deusto.client.windows.CreateOrganizerWindow.actionPerformed (
   (ActionEvent e )
```

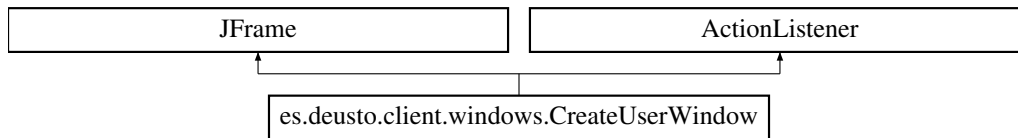
The documentation for this class was generated from the following file:

- `src/main/java/es/deusto/client/windows/`[CreateOrganizerWindow.java](#)

7.8 es.deusto.client.windows.CreateUserWindow Class Reference

This class is the one used to create a window that allows to create a new user.

Inheritance diagram for es.deusto.client.windows.CreateUserWindow:



Public Member Functions

- `CreateUserWindow` (`Controller` controller)
- `void actionPerformed` (`ActionEvent` e)

7.8.1 Detailed Description

This class is the one used to create a window that allows to create a new user.

7.8.2 Constructor & Destructor Documentation

7.8.2.1 CreateUserWindow()

```
es.deusto.client.windows.CreateUserWindow.CreateUserWindow (  
    Controller controller )
```

7.8.3 Member Function Documentation

7.8.3.1 actionPerformed()

```
void es.deusto.client.windows.CreateUserWindow.actionPerformed (  
    ActionEvent e )
```

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

- `src/main/java/es/deusto/client/windows/`[CreateUserWindow.java](#)

7.9 es.deusto.server.dao.DAOFactory Class Reference

This class is used to create a DAO factory that is used by all of the other DAO classes, to create them or to close them.

Public Member Functions

- [UserDAO](#) [createUserDAO](#) ()
- [OrganizerDAO](#) [createOrganizerDAO](#) ()
- [EventDAO](#) [createEventDAO](#) ()
- [TopicDAO](#) [createTopicDAO](#) ()
- [PostDAO](#) [createPostDAO](#) ()
- void [closeDAO](#) ([PostDAO](#) dao)
- void [closeDAO](#) ([TopicDAO](#) dao)
- void [closeDAO](#) ([EventDAO](#) dao)
- void [closeDAO](#) ([OrganizerDAO](#) dao)
- void [closeDAO](#) ([UserDAO](#) dao)

Static Public Member Functions

- static [DAOFactory](#) [getInstance](#) ()

7.9.1 Detailed Description

This class is used to create a DAO factory that is used by all of the other DAO classes, to create them or to close them.

7.9.2 Member Function Documentation

7.9.2.1 [closeDAO\(\)](#) [1/5]

```
void es.deusto.server.dao.DAOFactory.closeDAO (  
    EventDAO dao )
```

7.9.2.2 [closeDAO\(\)](#) [2/5]

```
void es.deusto.server.dao.DAOFactory.closeDAO (  
    OrganizerDAO dao )
```

7.9.2.3 closeDAO() [3/5]

```
void es.deusto.server.dao.DAOFactory.closeDAO (
    PostDAO dao )
```

7.9.2.4 closeDAO() [4/5]

```
void es.deusto.server.dao.DAOFactory.closeDAO (
    TopicDAO dao )
```

7.9.2.5 closeDAO() [5/5]

```
void es.deusto.server.dao.DAOFactory.closeDAO (
    UserDAO dao )
```

7.9.2.6 createEventDAO()

```
EventDAO es.deusto.server.dao.DAOFactory.createEventDAO ( )
```

7.9.2.7 createOrganizerDAO()

```
OrganizerDAO es.deusto.server.dao.DAOFactory.createOrganizerDAO ( )
```

7.9.2.8 createPostDAO()

```
PostDAO es.deusto.server.dao.DAOFactory.createPostDAO ( )
```

7.9.2.9 createTopicDAO()

```
TopicDAO es.deusto.server.dao.DAOFactory.createTopicDAO ( )
```

7.9.2.10 createUserDAO()

```
UserDAO es.deusto.server.dao.DAOFactory.createUserDAO ( )
```

7.9.2.11 getInstance()

```
static DAOFactory es.deusto.server.dao.DAOFactory.getInstance ( ) [static]
```

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

- src/main/java/es/deusto/server/dao/DAOFactory.java

7.10 es.deusto.server.data.Event Class Reference

This classes is the one that defines an event and it is used to create one.

Public Member Functions

- [Event](#) ()
- [Event](#) ([EventInfo](#) info)
- String [getName](#) ()
- void [setName](#) (String name)
- String [getDescription](#) ()
- void [setDescription](#) (String desc)
- [Topic](#) [getTopic](#) ()
- void [setTopic](#) ([Topic](#) topic)
- [Organizer](#) [getOrganizer](#) ()
- void [setOrganizer](#) ([Organizer](#) organizer)
- String [getCity](#) ()
- void [setCity](#) (String city)
- String [getCountry](#) ()
- void [setCountry](#) (String country)
- Date [getDate](#) ()
- void [setDate](#) (Date date)
- int [getInterested](#) ()
- void [setInterested](#) (int interested)
- void [addInterested](#) ()
- void [reduceInterested](#) ()
- ArrayList< [Post](#) > [getPosts](#) ()
- void [setPosts](#) (ArrayList< [Post](#) > posts)
- String [toString](#) ()

7.10.1 Detailed Description

This classes is the one that defines an event and it is used to create one.

7.10.2 Constructor & Destructor Documentation

7.10.2.1 Event() [1/2]

```
es.deusto.server.data.Event.Event ( )
```

7.10.2.2 Event() [2/2]

```
es.deusto.server.data.Event.Event (
    EventInfo info )
```

7.10.3 Member Function Documentation

7.10.3.1 addInterested()

```
void es.deusto.server.data.Event.addInterested ( )
```

7.10.3.2 getCity()

```
String es.deusto.server.data.Event.getCity ( )
```

7.10.3.3 getCountry()

```
String es.deusto.server.data.Event.getCountry ( )
```

7.10.3.4 getDate()

```
Date es.deusto.server.data.Event.getDate ( )
```

7.10.3.5 getDescription()

```
String es.deusto.server.data.Event.getDescription ( )
```

7.10.3.6 getInterested()

```
int es.deusto.server.data.Event.getInterested ( )
```

7.10.3.7 getName()

```
String es.deusto.server.data.Event.getName ( )
```

7.10.3.8 getOrganizer()

```
Organizer es.deusto.server.data.Event.getOrganizer ( )
```

7.10.3.9 getPosts()

```
ArrayList<Post> es.deusto.server.data.Event.getPosts ( )
```

7.10.3.10 getTopic()

```
Topic es.deusto.server.data.Event.getTopic ( )
```

7.10.3.11 reduceInterested()

```
void es.deusto.server.data.Event.reduceInterested ( )
```

7.10.3.12 setCity()

```
void es.deusto.server.data.Event.setCity (
    String city )
```

7.10.3.13 setCountry()

```
void es.deusto.server.data.Event.setCountry (
    String country )
```

7.10.3.14 setDate()

```
void es.deusto.server.data.Event.setDate (
    Date date )
```

7.10.3.15 setDescription()

```
void es.deusto.server.data.Event.setDescription (
    String desc )
```

7.10.3.16 setInterested()

```
void es.deusto.server.data.Event.setInterested (
    int interested )
```

7.10.3.17 setName()

```
void es.deusto.server.data.Event.setName (
    String name )
```

7.10.3.18 setOrganizer()

```
void es.deusto.server.data.Event.setOrganizer (
    Organizer organizer )
```

7.10.3.19 setPosts()

```
void es.deusto.server.data.Event.setPosts (
    ArrayList< Post > posts )
```

7.10.3.20 setTopic()

```
void es.deusto.server.data.Event.setTopic (
    Topic topic )
```

7.10.3.21 toString()

```
String es.deusto.server.data.Event.toString ( )
```

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

- src/main/java/es/deusto/server/data/[Event.java](#)

7.11 es.deusto.server.dao.EventDAO Class Reference

This classes is the one used to create an event DAO.

Public Member Functions

- `ArrayList< Event > getEvents (String name)`
Retrieves List of events from the database given the full name or part of it.
- `ArrayList< Event > getEventsbyUser (User user)`
Retrieves List of events from the database that match the user's interests and location.
- `ArrayList< Event > getEventsByOrganizer (Organizer organizer)`
Retrieves List of the events of a specific Organizer.
- `ArrayList< Event > getEvents ()`
Retrieves a list of all the events from the database.
- `void storeEvent (Event event)`
Stores a event in the database.
- `void updateEvent (Event event)`
Updates an event.
- `void deleteEvent (Event event)`
deletes an event from the database.

Protected Member Functions

- `EventDAO (PersistenceManager pm)`
- `PersistenceManager getPersistenceManager ()`
- `void close ()`

7.11.1 Detailed Description

This classes is the one used to create an event DAO.

7.11.2 Constructor & Destructor Documentation

7.11.2.1 EventDAO()

```
es.deusto.server.dao.EventDAO.EventDAO (
    PersistenceManager pm ) [protected]
```

7.11.3 Member Function Documentation

7.11.3.1 close()

```
void es.deusto.server.dao.EventDAO.close ( ) [protected]
```

7.11.3.2 deleteEvent()

```
void es.deusto.server.dao.EventDAO.deleteEvent (
    Event event )
```

deletes an event from the database.

Parameters

<i>event</i>	event to be deleted
--------------	---------------------

7.11.3.3 getEvents() [1/2]

```
ArrayList<Event> es.deusto.server.dao.EventDAO.getEvents ( )
```

Retrieves a list of all the events from the database.

7.11.3.4 getEvents() [2/2]

```
ArrayList<Event> es.deusto.server.dao.EventDAO.getEvents (
    String name )
```

Retrieves List of events from the database given the full name or part of it.

Parameters

<i>name</i>	name of the event
-------------	-------------------

Returns

list of events from the database

7.11.3.5 getEventsByOrganizer()

```
ArrayList<Event> es.deusto.server.dao.EventDAO.getEventsByOrganizer (
    Organizer organizer )
```

Retrieves List of the events of a specific Organizer.

Parameters

<i>organizer</i>	
------------------	--

Returns

list of events from the database

7.11.3.6 getEventsbyUser()

```
ArrayList<Event> es.deusto.server.dao.EventDAO.getEventsbyUser (
    User user )
```

Retrieves List of events from the database that match the user's interests and location.

this is use to create recomendation lists.

Parameters

<i>user</i>	User
-------------	------

Returns

list of events from the database

7.11.3.7 getPersistenceManager()

```
PersistenceManager es.deusto.server.dao.EventDAO.getPersistenceManager ( ) [protected]
```

7.11.3.8 storeEvent()

```
void es.deusto.server.dao.EventDAO.storeEvent (
    Event event )
```

Stores a event in the database.

Parameters

<i>event</i>	event to be submitted
--------------	-----------------------

7.11.3.9 updateEvent()

```
void es.deusto.server.dao.EventDAO.updateEvent (
    Event event )
```

Updates an event.

Parameters

<i>event</i>	
--------------	--

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

- `src/main/java/es/deusto/server/dao/EventDAO.java`

7.12 es.deusto.serialization.EventInfo Class Reference

DTO equivalent class of the Event class from the server.

Public Member Functions

- [EventInfo](#) ()
- [EventInfo](#) (Event e)
- String [getName](#) ()
- void [setName](#) (String name)
- String [getDescription](#) ()
- void [setDescription](#) (String description)

- [TopicInfo](#) [getTopic](#) ()
- void [setTopic](#) ([TopicInfo](#) topic)
- String [getOrganizerEmail](#) ()
- void [setOrganizerEmail](#) (String organizerEmail)
- ArrayList< [PostInfo](#) > [getPosts](#) ()
- void [setPosts](#) (ArrayList< [PostInfo](#) > posts)
- int [getInterested](#) ()
- void [setInterested](#) (int interested)
- void [addInterested](#) ()
- void [reduceInterested](#) ()
- String [getCity](#) ()
- void [setCity](#) (String city)
- String [getCountry](#) ()
- void [setCountry](#) (String country)
- Date [getDate](#) ()
- void [setDate](#) (Date date)

7.12.1 Detailed Description

DTO equivalent class of the Event class from the server.

This class is used to transport all the relevant information of an event through the network.

Since

Sprint 1

7.12.2 Constructor & Destructor Documentation

7.12.2.1 [EventInfo](#)() [1/2]

```
es.deusto.serialization.EventInfo.EventInfo ( )
```

7.12.2.2 [EventInfo](#)() [2/2]

```
es.deusto.serialization.EventInfo.EventInfo (
    Event e )
```

7.12.3 Member Function Documentation

7.12.3.1 addInterested()

```
void es.deusto.serialization.EventInfo.addInterested ( )
```

7.12.3.2 getCity()

```
String es.deusto.serialization.EventInfo.getCity ( )
```

7.12.3.3 getCountry()

```
String es.deusto.serialization.EventInfo.getCountry ( )
```

7.12.3.4 getDate()

```
Date es.deusto.serialization.EventInfo.getDate ( )
```

7.12.3.5 getDescription()

```
String es.deusto.serialization.EventInfo.getDescription ( )
```

7.12.3.6 getInterested()

```
int es.deusto.serialization.EventInfo.getInterested ( )
```

7.12.3.7 getName()

```
String es.deusto.serialization.EventInfo.getName ( )
```

7.12.3.8 getOrganizerEmail()

```
String es.deusto.serialization.EventInfo.getOrganizerEmail ( )
```

7.12.3.9 getPosts()

```
ArrayList<PostInfo> es.deusto.serialization.EventInfo.getPosts ( )
```

7.12.3.10 getTopic()

```
TopicInfo es.deusto.serialization.EventInfo.getTopic ( )
```

7.12.3.11 reduceInterested()

```
void es.deusto.serialization.EventInfo.reduceInterested ( )
```

7.12.3.12 setCity()

```
void es.deusto.serialization.EventInfo.setCity (
    String city )
```

7.12.3.13 setCountry()

```
void es.deusto.serialization.EventInfo.setCountry (
    String country )
```

7.12.3.14 setDate()

```
void es.deusto.serialization.EventInfo.setDate (
    Date date )
```

7.12.3.15 setDescription()

```
void es.deusto.serialization.EventInfo.setDescription (
    String description )
```

7.12.3.16 setInterested()

```
void es.deusto.serialization.EventInfo.setInterested (
    int interested )
```

7.12.3.17 setName()

```
void es.deusto.serialization.EventInfo.setName (
    String name )
```

7.12.3.18 setOrganizerEmail()

```
void es.deusto.serialization.EventInfo.setOrganizerEmail (
    String organizerEmail )
```

7.12.3.19 setPosts()

```
void es.deusto.serialization.EventInfo.setPosts (
    ArrayList< PostInfo > posts )
```

7.12.3.20 setTopic()

```
void es.deusto.serialization.EventInfo.setTopic (
    TopicInfo topic )
```

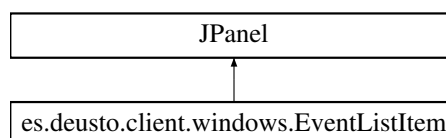
The documentation for this class was generated from the following file:

- [src/main/java/es/deusto/serialization/EventInfo.java](#)

7.13 es.deusto.client.windows.EventListItem Class Reference

This class is the one used to create a window that allows the user to have an event list.

Inheritance diagram for es.deusto.client.windows.EventListItem:



Public Member Functions

- [EventListItem](#) ([Controller](#) controller, [EventInfo](#) event)

7.13.1 Detailed Description

This class is the one used to create a window that allows the user to have an event list.

7.13.2 Constructor & Destructor Documentation

7.13.2.1 EventListItem()

```
es.deusto.client.windows.EventListItem.EventListItem (
    Controller controller,
    EventInfo event )
```

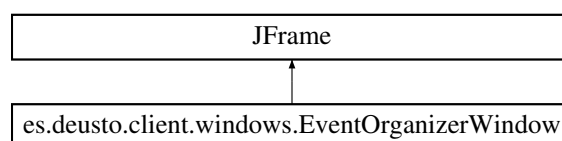
The documentation for this class was generated from the following file:

- [src/main/java/es/deusto/client/windows/EventListItem.java](#)

7.14 es.deusto.client.windows.EventOrganizerWindow Class Reference

This class is the one used to create a window that allows to an organizer to change or administrate one of his events.

Inheritance diagram for es.deusto.client.windows.EventOrganizerWindow:



Public Member Functions

- [EventOrganizerWindow](#) ([Controller](#) controller)

7.14.1 Detailed Description

This class is the one used to create a window that allows to an organizer to change or administrate one of his events.

7.14.2 Constructor & Destructor Documentation

7.14.2.1 EventOrganizerWindow()

```
es.deusto.client.windows.EventOrganizerWindow.EventOrganizerWindow (
    Controller controller )
```

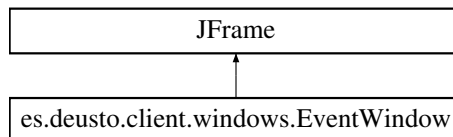
The documentation for this class was generated from the following file:

- src/main/java/es/deusto/client/windows/[EventOrganizerWindow.java](#)

7.15 es.deusto.client.windows.EventWindow Class Reference

This class is the one used to create a window that allows to display all the information of one event.

Inheritance diagram for es.deusto.client.windows.EventWindow:



Public Member Functions

- [EventWindow](#) ([Controller](#) controller, [EventInfo](#) e)

7.15.1 Detailed Description

This class is the one used to create a window that allows to display all the information of one event.

7.15.2 Constructor & Destructor Documentation

7.15.2.1 EventWindow()

```
es.deusto.client.windows.EventWindow.EventWindow (
    Controller controller,
    EventInfo e )
```

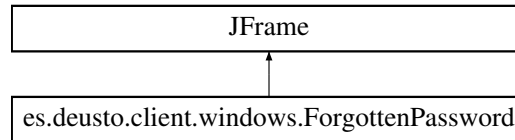
The documentation for this class was generated from the following file:

- src/main/java/es/deusto/client/windows/[EventWindow.java](#)

7.16 es.deusto.client.windows.ForgottenPassword Class Reference

This class is the one used to create a window that allows a user to log in when he forgets the password.

Inheritance diagram for es.deusto.client.windows.ForgottenPassword:



Public Member Functions

- [ForgottenPassword](#) ([Controller](#) controller)
Create the frame.

7.16.1 Detailed Description

This class is the one used to create a window that allows a user to log in when he forgets the password.

7.16.2 Constructor & Destructor Documentation

7.16.2.1 ForgottenPassword()

```
es.deusto.client.windows.ForgottenPassword.ForgottenPassword (
    Controller controller )
```

Create the frame.

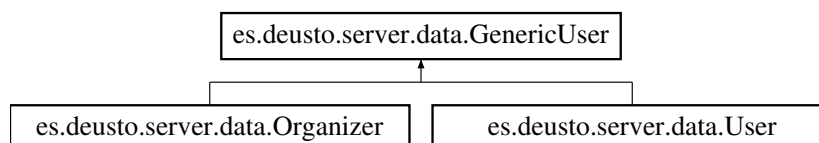
The documentation for this class was generated from the following file:

- `src/main/java/es/deusto/client/windows/ForgottenPassword.java`

7.17 es.deusto.server.data.GenericUser Class Reference

This classes is the father class for normal user and organizer.

Inheritance diagram for es.deusto.server.data.GenericUser:



Public Member Functions

- String [getName](#) ()
- void [setName](#) (String name)
- String [getEmail](#) ()
- void [setEmail](#) (String email)
- String [getPassword](#) ()
- void [setPassword](#) (String password)
- String [toString](#) ()

7.17.1 Detailed Description

This classes is the father class for normal user and organizer.

Both of them are users so they inherit from this class.

7.17.2 Member Function Documentation

7.17.2.1 getEmail()

```
String es.deusto.server.data.GenericUser.getEmail ( )
```

7.17.2.2 getName()

```
String es.deusto.server.data.GenericUser.getName ( )
```

7.17.2.3 getPassword()

```
String es.deusto.server.data.GenericUser.getPassword ( )
```

7.17.2.4 setEmail()

```
void es.deusto.server.data.GenericUser.setEmail (
    String email )
```

7.17.2.5 setName()

```
void es.deusto.server.data.GenericUser.setName (
    String name )
```

7.17.2.6 setPassword()

```
void es.deusto.server.data.GenericUser.setPassword (
    String password )
```

7.17.2.7 toString()

```
String es.deusto.server.data.GenericUser.toString ( )
```

Reimplemented in [es.deusto.server.data.User](#), and [es.deusto.server.data.Organizer](#).

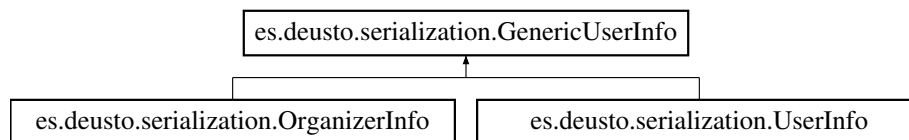
The documentation for this class was generated from the following file:

- [src/main/java/es/deusto/server/data/GenericUser.java](#)

7.18 es.deusto.serialization.GenericUserInfo Class Reference

This class works as a Tag for the Classes related to the User and Organizer.

Inheritance diagram for es.deusto.serialization.GenericUserInfo:



7.18.1 Detailed Description

This class works as a Tag for the Classes related to the User and Organizer.

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

- [src/main/java/es/deusto/serialization/GenericUserInfo.java](#)

7.19 es.deusto.client.windows.LanguageManager Class Reference

This class is used to manage the different languages supported by the application.

Public Member Functions

- [LanguageManager](#) ()
Constructor.
- [LanguageManager](#) (String language)
Creates an instance of the class and sets the current language to the specified one.
- String [getString](#) (String key)
Retrieves the translation to the current language of the String identified by the key.
- String [getLanguage](#) ()
- void [setLanguage](#) (String language)
Sets the current language.

7.19.1 Detailed Description

This class is used to manage the different languages supported by the application.

It keeps track of the current language, and it allows the users to change it and to get the specified string in the current language.

7.19.2 Constructor & Destructor Documentation

7.19.2.1 LanguageManager() [1/2]

```
es.deusto.client.windows.LanguageManager.LanguageManager ( )
```

Constructor.

Loads the default system language.

7.19.2.2 LanguageManager() [2/2]

```
es.deusto.client.windows.LanguageManager.LanguageManager (
    String language )
```

Creates an instance of the class and sets the current language to the specified one.

Parameters

<i>language</i>	The String code of the desired language
-----------------	---

7.19.3 Member Function Documentation

7.19.3.1 `getLanguage()`

```
String es.deusto.client.windows.LanguageManager.getLanguage ( )
```

7.19.3.2 `getString()`

```
String es.deusto.client.windows.LanguageManager.getString (
    String key )
```

Retrieves the translation to the current language of the String identified by the key.

Parameters

<i>key</i>	Identifier of the desired String
------------	----------------------------------

Returns

translation Translation of the specified String to the current language

7.19.3.3 `setLanguage()`

```
void es.deusto.client.windows.LanguageManager.setLanguage (
    String language )
```

Sets the current language.

Parameters

<i>language</i>	The String code of the desired language
-----------------	---

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

- [src/main/java/es/deusto/client/windows/LanguageManager.java](#)

7.20 `es.deusto.serialization.LoginAttempt` Class Reference

This class encapsulates all the information required when a user (Either normal User or Organizer) logs in.

Public Member Functions

- [LoginAttempt](#) ()
- [LoginAttempt](#) (String email, String password, boolean organizer)
Constructor of a Log in attempt.
- String [getEmail](#) ()
- void [setEmail](#) (String email)
- String [getPassword](#) ()
- void [setPassword](#) (String password)
- boolean [isOrganizer](#) ()
- void [setOrganizer](#) (boolean organizer)
- String [toString](#) ()

7.20.1 Detailed Description

This class encapsulates all the information required when a user (Either normal User or Organizer) logs in.

Since

Sprint 1

7.20.2 Constructor & Destructor Documentation

7.20.2.1 LoginAttempt() [1/2]

```
es.deusto.serialization.LoginAttempt.LoginAttempt ( )
```

7.20.2.2 LoginAttempt() [2/2]

```
es.deusto.serialization.LoginAttempt.LoginAttempt (
    String email,
    String password,
    boolean organizer )
```

Constructor of a Log in attempt.

Encapsulates the information returned by the server when a user (Either normal User or Organizer) Logs in.

Parameters

<i>email</i>	email of the user that will log.
<i>password</i>	password of the account.
<i>organizer</i>	boolean parameter that states if this user is an organizer or not.

7.20.3 Member Function Documentation

7.20.3.1 getEmail()

```
String es.deusto.serialization.LoginAttempt.getEmail ( )
```

7.20.3.2 getPassword()

```
String es.deusto.serialization.LoginAttempt.getPassword ( )
```

7.20.3.3 isOrganizer()

```
boolean es.deusto.serialization.LoginAttempt.isOrganizer ( )
```

7.20.3.4 setEmail()

```
void es.deusto.serialization.LoginAttempt.setEmail (
    String email )
```

7.20.3.5 setOrganizer()

```
void es.deusto.serialization.LoginAttempt.setOrganizer (
    boolean organizer )
```

7.20.3.6 setPassword()

```
void es.deusto.serialization.LoginAttempt.setPassword (
    String password )
```


7.20.3.7 toString()

```
String es.deusto.serialization.LoginAttempt.toString ( )
```

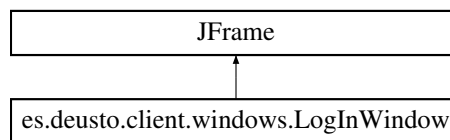
The documentation for this class was generated from the following file:

- [src/main/java/es/deusto/serialization/LoginAttempt.java](#)

7.21 es.deusto.client.windows.LogInWindow Class Reference

This class is the one used to create a window that allows a user/organizer to log in to the application.

Inheritance diagram for es.deusto.client.windows.LogInWindow:



Public Member Functions

- [LogInWindow](#) ([Controller](#) controller)

Public Attributes

- `TextField` [id](#)
- `Button` [bCreate](#)
- `CheckBox` [cBoxAdmin](#)

7.21.1 Detailed Description

This class is the one used to create a window that allows a user/organizer to log in to the application.

7.21.2 Constructor & Destructor Documentation

7.21.2.1 LogInWindow()

```
es.deusto.client.windows.LogInWindow.LogInWindow (
    Controller controller )
```

7.21.3 Member Data Documentation

7.21.3.1 bCreate

`JButton es.deusto.client.windows.LogInWindow.bCreate`

7.21.3.2 cBoxAdmin

`JCheckBox es.deusto.client.windows.LogInWindow.cBoxAdmin`

7.21.3.3 id

`JTextField es.deusto.client.windows.LogInWindow.id`

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

- `src/main/java/es/deusto/client/windows/LogInWindow.java`

7.22 es.deusto.client.main.Main Class Reference

This class is the one that starts the application.

Static Public Member Functions

- static void `main` (String[] args) throws `ClassNotFoundException`

7.22.1 Detailed Description

This class is the one that starts the application.

It creates a new controller for the user to connect to the server and opens a new `LogInWindow` where the user can start using the application by login in or creating a new user.

7.22.2 Member Function Documentation

7.22.2.1 main()

```
static void es.deusto.client.main.Main.main (
    String[] args ) throws ClassNotFoundException [static]
```

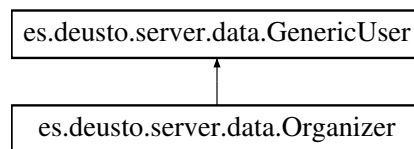
The documentation for this class was generated from the following file:

- src/main/java/es/deusto/client/main/[Main.java](#)

7.23 es.deusto.server.data.Organizer Class Reference

This classes is the one that defines an organizer and it is used to create one.

Inheritance diagram for es.deusto.server.data.Organizer:



Public Member Functions

- [Organizer](#) ()
- String [getOrganization](#) ()
- void [setOrganization](#) (String organization)
- ArrayList< [Event](#) > [getCreatedEvents](#) ()
- void [setCreatedEvents](#) (ArrayList< [Event](#) > createdEvents)
- void [addCreatedEvent](#) ([Event](#) e)
- String [toString](#) ()

7.23.1 Detailed Description

This classes is the one that defines an organizer and it is used to create one.

7.23.2 Constructor & Destructor Documentation

7.23.2.1 Organizer()

```
es.deusto.server.data.Organizer.Organizer ( )
```

7.23.3 Member Function Documentation

7.23.3.1 addCreatedEvent()

```
void es.deusto.server.data.Organizer.addCreatedEvent (
    Event e )
```

7.23.3.2 getCreatedEvents()

```
ArrayList<Event> es.deusto.server.data.Organizer.getCreatedEvents ( )
```

7.23.3.3 getOrganization()

```
String es.deusto.server.data.Organizer.getOrganization ( )
```

7.23.3.4 setCreatedEvents()

```
void es.deusto.server.data.Organizer.setCreatedEvents (
    ArrayList< Event > createdEvents )
```

Parameters

<i>createdEvents</i>	the createdEvents to set
----------------------	--------------------------

7.23.3.5 setOrganization()

```
void es.deusto.server.data.Organizer.setOrganization (
    String organization )
```

7.23.3.6 toString()

```
String es.deusto.server.data.Organizer.toString ( )
```

Reimplemented from [es.deusto.server.data.GenericUser](#).

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

- [src/main/java/es/deusto/server/data/Organizer.java](#)

7.24 es.deusto.server.dao.OrganizerDAO Class Reference

This classes is the one used to create an organizer DAO.

Public Member Functions

- [Organizer getOrganizer](#) (String email)
- void [storeOrganizer](#) ([Organizer](#) organizer)
- void [updateOrganizer](#) ([Organizer](#) organizer)
- void [deleteOrganizer](#) (String email)

Protected Member Functions

- [OrganizerDAO](#) (PersistenceManager pm)
- PersistenceManager [getPersistenceManager](#) ()
- void [close](#) ()

7.24.1 Detailed Description

This classes is the one used to create an organizer DAO.

7.24.2 Constructor & Destructor Documentation

7.24.2.1 OrganizerDAO()

```
es.deusto.server.dao.OrganizerDAO.OrganizerDAO (  
    PersistenceManager pm ) [protected]
```

7.24.3 Member Function Documentation

7.24.3.1 close()

```
void es.deusto.server.dao.OrganizerDAO.close ( ) [protected]
```

7.24.3.2 deleteOrganizer()

```
void es.deusto.server.dao.OrganizerDAO.deleteOrganizer (
    String email )
```

7.24.3.3 getOrganizer()

```
Organizer es.deusto.server.dao.OrganizerDAO.getOrganizer (
    String email )
```

7.24.3.4 getPersistenceManager()

```
PersistenceManager es.deusto.server.dao.OrganizerDAO.getPersistenceManager ( ) [protected]
```

7.24.3.5 storeOrganizer()

```
void es.deusto.server.dao.OrganizerDAO.storeOrganizer (
    Organizer organizer )
```

7.24.3.6 updateOrganizer()

```
void es.deusto.server.dao.OrganizerDAO.updateOrganizer (
    Organizer organizer )
```

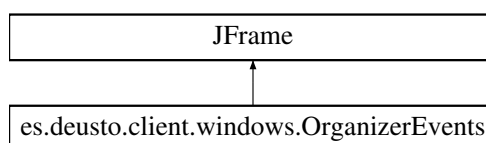
The documentation for this class was generated from the following file:

- [src/main/java/es/deusto/server/dao/OrganizerDAO.java](#)

7.25 es.deusto.client.windows.OrganizerEvents Class Reference

This class is the one used to create a window that allows an organizer to have all of his events.

Inheritance diagram for es.deusto.client.windows.OrganizerEvents:



Public Member Functions

- [OrganizerEvents](#) ([Controller](#) controller)

7.25.1 Detailed Description

This class is the one used to create a window that allows an organizer to have all of his events.

7.25.2 Constructor & Destructor Documentation

7.25.2.1 OrganizerEvents()

```
es.deusto.client.windows.OrganizerEvents.OrganizerEvents (
    Controller controller )
```

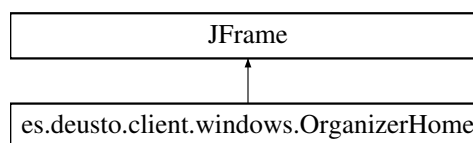
The documentation for this class was generated from the following file:

- [src/main/java/es/deusto/client/windows/OrganizerEvents.java](#)

7.26 es.deusto.client.windows.OrganizerHome Class Reference

This class is the one used to create a window that allows an organizer to have a home window where he can access to all of the functionalities of the application.

Inheritance diagram for es.deusto.client.windows.OrganizerHome:



Public Member Functions

- [OrganizerHome](#) ([Controller](#) controller)

7.26.1 Detailed Description

This class is the one used to create a window that allows an organizer to have a home window where he can access to all of the functionalities of the application.

7.26.2 Constructor & Destructor Documentation

7.26.2.1 OrganizerHome()

```
es.deusto.client.windows.OrganizerHome.OrganizerHome (
    Controller controller )
```

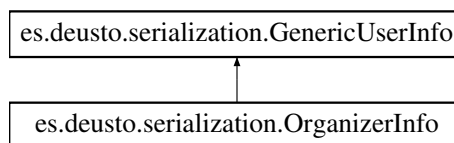
The documentation for this class was generated from the following file:

- [src/main/java/es/deusto/client/windows/OrganizerHome.java](#)

7.27 es.deusto.serialization.OrganizerInfo Class Reference

DTO equivalent class of the Organizer class from the server.

Inheritance diagram for es.deusto.serialization.OrganizerInfo:



Public Member Functions

- [OrganizerInfo](#) ()
- [OrganizerInfo](#) ([Organizer](#) organizer)
- String [getName](#) ()
- void [setName](#) (String name)
- String [getEmail](#) ()
- void [setEmail](#) (String email)
- String [getOrganization](#) ()
- void [setOrganization](#) (String organization)
- ArrayList< [EventInfo](#) > [getCreatedEvents](#) ()
- void [setCreatedEvents](#) (ArrayList< [EventInfo](#) > createdEvents)

7.27.1 Detailed Description

DTO equivalent class of the Organizer class from the server.

This class is used to transport all the relevant information of an organizer through the network.

Since

Sprint 1

7.27.2 Constructor & Destructor Documentation

7.27.2.1 OrganizerInfo() [1/2]

```
es.deusto.serialization.OrganizerInfo.OrganizerInfo ( )
```

7.27.2.2 OrganizerInfo() [2/2]

```
es.deusto.serialization.OrganizerInfo.OrganizerInfo (
    Organizer organizer )
```

7.27.3 Member Function Documentation

7.27.3.1 getCreatedEvents()

```
ArrayList<EventInfo> es.deusto.serialization.OrganizerInfo.getCreatedEvents ( )
```

7.27.3.2 getEmail()

```
String es.deusto.serialization.OrganizerInfo.getEmail ( )
```

7.27.3.3 getName()

```
String es.deusto.serialization.OrganizerInfo.getName ( )
```

7.27.3.4 getOrganization()

```
String es.deusto.serialization.OrganizerInfo.getOrganization ( )
```

7.27.3.5 setCreatedEvents()

```
void es.deusto.serialization.OrganizerInfo.setCreatedEvents (
    ArrayList< EventInfo > createdEvents )
```

7.27.3.6 setEmail()

```
void es.deusto.serialization.OrganizerInfo.setEmail (
    String email )
```

7.27.3.7 setName()

```
void es.deusto.serialization.OrganizerInfo.setName (
    String name )
```

7.27.3.8 setOrganization()

```
void es.deusto.serialization.OrganizerInfo.setOrganization (
    String organization )
```

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

- [src/main/java/es/deusto/serialization/OrganizerInfo.java](#)

7.28 es.deusto.server.data.Post Class Reference

This classes is the one that defines a post and it is used to create one.

Public Member Functions

- [Post](#) ()
- [Post](#) ([PostInfo](#) info)
- String [getTitle](#) ()
- void [setTitle](#) (String title)
- String [getDescription](#) ()
- void [setDescription](#) (String description)
- Date [getDate](#) ()
- void [setDate](#) (Date date)
- String [getEventName](#) ()
- void [setEventName](#) (String eventName)
- String [getOrganizerEmail](#) ()
- void [setOrganizerEmail](#) (String organizerEmail)

7.28.1 Detailed Description

This classes is the one that defines a post and it is used to create one.

7.28.2 Constructor & Destructor Documentation

7.28.2.1 Post() [1/2]

```
es.deusto.server.data.Post.Post ( )
```

7.28.2.2 Post() [2/2]

```
es.deusto.server.data.Post.Post (
    PostInfo info )
```

7.28.3 Member Function Documentation

7.28.3.1 getDate()

```
Date es.deusto.server.data.Post.getDate ( )
```

7.28.3.2 getDescription()

```
String es.deusto.server.data.Post.getDescription ( )
```

7.28.3.3 getEventName()

```
String es.deusto.server.data.Post.getEventName ( )
```

7.28.3.4 getOrganizerEmail()

```
String es.deusto.server.data.Post.getOrganizerEmail ( )
```

7.28.3.5 getTitle()

```
String es.deusto.server.data.Post.getTitle ( )
```

7.28.3.6 setDate()

```
void es.deusto.server.data.Post.setDate (
    Date date )
```

7.28.3.7 setDescription()

```
void es.deusto.server.data.Post.setDescription (
    String description )
```

7.28.3.8 setEventName()

```
void es.deusto.server.data.Post.setEventName (
    String eventName )
```

7.28.3.9 setOrganizerEmail()

```
void es.deusto.server.data.Post.setOrganizerEmail (
    String organizerEmail )
```

7.28.3.10 setTitle()

```
void es.deusto.server.data.Post.setTitle (
    String title )
```

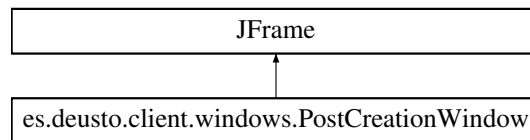
The documentation for this class was generated from the following file:

- src/main/java/es/deusto/server/data/[Post.java](#)

7.29 es.deusto.client.windows.PostCreationWindow Class Reference

This class is the one used to create a window that allows an organizer to have like a double check of the event he has just created.

Inheritance diagram for es.deusto.client.windows.PostCreationWindow:



Public Member Functions

- [PostCreationWindow](#) ([Controller](#) controller, [EventInfo](#) event)

7.29.1 Detailed Description

This class is the one used to create a window that allows an organizer to have like a double check of the event he has just created.

7.29.2 Constructor & Destructor Documentation

7.29.2.1 PostCreationWindow()

```
es.deusto.client.windows.PostCreationWindow.PostCreationWindow (
    Controller controller,
    EventInfo event )
```

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

- src/main/java/es/deusto/client/windows/[PostCreationWindow.java](#)

7.30 es.deusto.server.dao.PostDAO Class Reference

This classes is the one used to create a DAO for a post.

Public Member Functions

- [ArrayList< Post >](#) [getPostsByEvent](#) ([Event](#) event)
- void [storePost](#) ([Post](#) post)
- void [updatePost](#) ([Post](#) post)
- void [deletePost](#) ([Post](#) post)

Protected Member Functions

- [PostDAO](#) (PersistenceManager pm)
- PersistenceManager [getPersistenceManager](#) ()
- void [close](#) ()

7.30.1 Detailed Description

This classes is the one used to create a DAO for a post.

7.30.2 Constructor & Destructor Documentation

7.30.2.1 PostDAO()

```
es.deusto.server.dao.PostDAO.PostDAO (  
    PersistenceManager pm ) [protected]
```

7.30.3 Member Function Documentation

7.30.3.1 close()

```
void es.deusto.server.dao.PostDAO.close ( ) [protected]
```

7.30.3.2 deletePost()

```
void es.deusto.server.dao.PostDAO.deletePost (  
    Post post )
```

7.30.3.3 getPersistenceManager()

```
PersistenceManager es.deusto.server.dao.PostDAO.getPersistenceManager ( ) [protected]
```

7.30.3.4 getPostsByEvent()

```
ArrayList<Post> es.deusto.server.dao.PostDAO.getPostsByEvent (
    Event event )
```

7.30.3.5 storePost()

```
void es.deusto.server.dao.PostDAO.storePost (
    Post post )
```

7.30.3.6 updatePost()

```
void es.deusto.server.dao.PostDAO.updatePost (
    Post post )
```

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

- [src/main/java/es/deusto/server/dao/PostDAO.java](#)

7.31 es.deusto.serialization.PostInfo Class Reference

DTO equivalent class of the Post class from the server.

Public Member Functions

- [PostInfo \(\)](#)
- [PostInfo \(Post post\)](#)
- String [getTitle \(\)](#)
- void [setTitle \(String title\)](#)
- String [getDescription \(\)](#)
- void [setDescription \(String description\)](#)
- Date [getDate \(\)](#)
- void [setDate \(Date date\)](#)
- String [getEventName \(\)](#)
- void [setEventName \(String eventName\)](#)
- String [getOrganizerEmail \(\)](#)
- void [setOrganizerEmail \(String organizerEmail\)](#)

7.31.1 Detailed Description

DTO equivalent class of the Post class from the server.

This class is used to transport all the relevant information of a post through the network.

Since

Sprint 2

7.31.2 Constructor & Destructor Documentation

7.31.2.1 PostInfo() [1/2]

```
es.deusto.serialization.PostInfo.PostInfo ( )
```

7.31.2.2 PostInfo() [2/2]

```
es.deusto.serialization.PostInfo.PostInfo (
    Post post )
```

7.31.3 Member Function Documentation

7.31.3.1 getDate()

```
Date es.deusto.serialization.PostInfo.getDate ( )
```

7.31.3.2 getDescription()

```
String es.deusto.serialization.PostInfo.getDescription ( )
```

7.31.3.3 getEventName()

```
String es.deusto.serialization.PostInfo.getEventName ( )
```

7.31.3.4 getOrganizerEmail()

```
String es.deusto.serialization.PostInfo.getOrganizerEmail ( )
```


7.31.3.5 getTitle()

```
String es.deusto.serialization.PostInfo.getTitle ( )
```

7.31.3.6 setDate()

```
void es.deusto.serialization.PostInfo.setDate (
    Date date )
```

7.31.3.7 setDescription()

```
void es.deusto.serialization.PostInfo.setDescription (
    String description )
```

7.31.3.8 setEventName()

```
void es.deusto.serialization.PostInfo.setEventName (
    String eventName )
```

7.31.3.9 setOrganizerEmail()

```
void es.deusto.serialization.PostInfo.setOrganizerEmail (
    String organizerEmail )
```

7.31.3.10 setTitle()

```
void es.deusto.serialization.PostInfo.setTitle (
    String title )
```

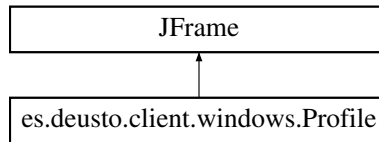
The documentation for this class was generated from the following file:

- [src/main/java/es/deusto/serialization/PostInfo.java](#)

7.32 es.deusto.client.windows.Profile Class Reference

This class is the one used to create a window that allows a user/organizer to have a profile with all of his information.

Inheritance diagram for es.deusto.client.windows.Profile:



Public Member Functions

- [Profile](#) ([Controller](#) controller, JFrame mainWindow)

7.32.1 Detailed Description

This class is the one used to create a window that allows a user/organizer to have a profile with all of his information.

7.32.2 Constructor & Destructor Documentation

7.32.2.1 Profile()

```
es.deusto.client.windows.Profile.Profile (
    Controller controller,
    JFrame mainWindow )
```

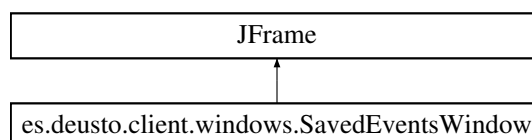
The documentation for this class was generated from the following file:

- [src/main/java/es/deusto/client/windows/Profile.java](#)

7.33 es.deusto.client.windows.SavedEventsWindow Class Reference

This class is the one used to create a window that allows to scroll, search and choose from all of the events stored in the database.

Inheritance diagram for es.deusto.client.windows.SavedEventsWindow:



Public Member Functions

- [SavedEventsWindow](#) ([Controller](#) controller)

7.33.1 Detailed Description

This class is the one used to create a window that allows to scroll, search and choose from all of the events stored in the database.

7.33.2 Constructor & Destructor Documentation

7.33.2.1 SavedEventsWindow()

```
es.deusto.client.windows.SavedEventsWindow.SavedEventsWindow (
    Controller controller )
```

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

- src/main/java/es/deusto/client/windows/[SavedEventsWindow.java](#)

7.34 es.deusto.junit.serialization.SerializationTest Class Reference

This classes is the one that tests functionalities such as creating an event, creating a user, orginzer, crea ting posts and topics.

Public Member Functions

- void [beforeClass](#) ()
- void [testEventInfoFromEvent](#) ()
- void [testEventInfoSetters](#) ()
- void [testOrganizerInfoFromOrganizer](#) ()
- void [testOrganizerInfoSetters](#) ()
- void [testPostInfoFromPost](#) ()
- void [testPostInfoSetters](#) ()
- void [testUserInfoFromUser](#) ()
- void [testUserInfoSetters](#) ()
- void [testUserInfoStringConstructor](#) ()
- void [testSignupAttemptUser](#) ()
- void [testSignupAttemptOrganizer](#) ()

Static Public Member Functions

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

7.34.1 Detailed Description

This classes is the one that tests functionalities such as creating an event, creating a user, orginzer, crea ting posts and topics.

Is the class that checks that all the functionalities work as intended.

7.34.2 Member Function Documentation

7.34.2.1 beforeClass()

```
void es.deusto.junit.serialization.SerializationTest.beforeClass ( )
```

7.34.2.2 startUp()

```
static void es.deusto.junit.serialization.SerializationTest.startUp ( ) [static]
```

7.34.2.3 suite()

```
static junit.framework.Test es.deusto.junit.serialization.SerializationTest.suite ( ) [static]
```

7.34.2.4 testEventInfoFromEvent()

```
void es.deusto.junit.serialization.SerializationTest.testEventInfoFromEvent ( )
```

7.34.2.5 testEventInfoSetters()

```
void es.deusto.junit.serialization.SerializationTest.testEventInfoSetters ( )
```

7.34.2.6 testOrganizerInfoFromOrganizer()

```
void es.deusto.junit.serialization.SerializationTest.testOrganizerInfoFromOrganizer ( )
```

7.34.2.7 testOrganizerInfoSetters()

```
void es.deusto.junit.serialization.SerializationTest.testOrganizerInfoSetters ( )
```

7.34.2.8 testPostInfoFromPost()

```
void es.deusto.junit.serialization.SerializationTest.testPostInfoFromPost ( )
```

7.34.2.9 testPostInfoSetters()

```
void es.deusto.junit.serialization.SerializationTest.testPostInfoSetters ( )
```

7.34.2.10 testSignupAttemptOrganizer()

```
void es.deusto.junit.serialization.SerializationTest.testSignupAttemptOrganizer ( )
```

7.34.2.11 testSignupAttemptUser()

```
void es.deusto.junit.serialization.SerializationTest.testSignupAttemptUser ( )
```

7.34.2.12 testUserInfoFromUser()

```
void es.deusto.junit.serialization.SerializationTest.testUserInfoFromUser ( )
```

7.34.2.13 testUserInfoSetters()

```
void es.deusto.junit.serialization.SerializationTest.testUserInfoSetters ( )
```

7.34.2.14 testUserInfoStringConstructor()

```
void es.deusto.junit.serialization.SerializationTest.testUserInfoStringConstructor ( )
```

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

- [src/test/java/es/deusto/junit/serialization/SerializationTest.java](#)

7.35 es.deusto.server.server.Server Class Reference

This class is the one that receives the requests from the client.

Public Member Functions

- [Server](#) ()
- Response [attemptNormalLogin](#) ([LoginAttempt](#) login)
This is used for the login process of regular users.
- Response [attemptOrganizerLogin](#) ([LoginAttempt](#) login)
This method is used for the login process of organizers.
- Response [attemptNormalSignup](#) ([SignupAttempt](#) signup)
This method is invoked whenever a POST request is made to the following path: /signup .
- Response [attemptOrganizerSignup](#) ([SignupAttempt](#) signup)
This method is used of the signup method of Organizers.
- Response [attemptNormalUpdate](#) ([SignupAttempt](#) signup)
This method is used to update the information of regular users' account.
- Response [attemptOrganizerUpdate](#) ([SignupAttempt](#) signup)
This method is used to update the information of an Organizer account.
- Response [getRecommendedEvents](#) ([SignupAttempt](#) signupAttempt)
This method is used for recommend Event to users.
- Response [recoverPassword](#) ([LoginAttempt](#) userInfo)
This method is used for password recovery for users who can't remember their passwords.
- Response [createEvent](#) ([EventInfo](#) eventInfo)
This method is used for the creation of new events from a given organizer.
- Response [createPost](#) ([PostInfo](#) postInfo)
This method is used for the creation of new Post from a given Event.
- Response [attemptUserDelete](#) ([UserInfo](#) userInfo)
This method is used of the delete method of regular Users.
- Response [attemptOrganizerDelete](#) ([OrganizerInfo](#) organizerInfo)
This method is used of the delete method of Organizers.
- Response [sayHello](#) ()
method used for manual testing

7.35.1 Detailed Description

This class is the one that receives the requests from the client.

There is no functionality implemented in this class, all the possible requests have a corresponding method inside the [AppService](#) class in which all the functionality is implemented. That way, the [Server](#) class is abstracted from the business objects.

7.35.2 Constructor & Destructor Documentation

7.35.2.1 Server()

```
es.deusto.server.server.Server.Server ( )
```

7.35.3 Member Function Documentation

7.35.3.1 attemptNormalLogin()

```
Response es.deusto.server.server.Server.attemptNormalLogin (
    LoginAttempt login )
```

This is used for the login process of regular users.

It is invoked whenever a POST request is made to the following path: /login. It is used for the login process of regular users.

Parameters

<i>login</i>	LoginAttempt class containing the information of the user needed for its log in (email and password).
--------------	---

Since

Sprint 1.

Returns

Response object with the information of the user, in case of valid credentials. If not, an empty response is returned.

7.35.3.2 attemptNormalSignup()

```
Response es.deusto.server.server.Server.attemptNormalSignup (
    SignupAttempt signup )
```

This method is invoked whenever a POST request is made to the following path: /signup .

It is used of the signup method of regular users.

Parameters

<i>signup</i>	Signup information of the user (name, email, password, city and interests) contained in a SignupAttempt object.
---------------	---

Returns

Response object with the information of the user, in case of correct user registration. If not, an empty response is returned

Since

Sprint 1

7.35.3.3 attemptNormalUpdate()

```
Response es.deusto.server.server.Server.attemptNormalUpdate (
    SignupAttempt signup )
```

This method is used to update the information of regular users' account.

It is invoked whenever a POST request is made to the following path: /update .

Parameters

<i>signup</i>	SignupAttempt object containing the information of the User (name, email, password, city and interests). In this case it is also used for updating their information since its basic structure doesn't change.
---------------	--

Since

Sprint 2

Returns

Response object with the information of the user, in case of correct user registration. If not, an empty response is returned

7.35.3.4 attemptOrganizerDelete()

```
Response es.deusto.server.server.Server.attemptOrganizerDelete (
    OrganizerInfo organizerInfo )
```

This method is used of the delete method of Organizers.

It is invoked whenever a POST request is made to the following path: /delete .

Parameters

<i>organizerInfo</i>	OrganizerInfo object containing Organizer's email.
----------------------	--

Since

Sprint 2

Returns

Response string with the Organizer's email, in case the delete was successful. If not, an empty response is returned.

7.35.3.5 attemptOrganizerLogin()

```
Response es.deusto.server.server.Server.attemptOrganizerLogin (
    LoginAttempt login )
```

This method is used for the login process of organizers.

It is invoked whenever a POST request is made to the following path: /loginOrganizer.

Parameters

<i>login</i>	LoginAttempt class containing the information of the organizer needed for its log in (email and password).
--------------	--

Returns

Response object with the information of the organizer, in case of valid credentials. If not, an empty response is returned.

Since

Sprint 2

7.35.3.6 attemptOrganizerSignup()

```
Response es.deusto.server.server.Server.attemptOrganizerSignup (
    SignupAttempt signup )
```

This method is used of the signup method of Organizers.

It is invoked whenever a POST request is made to the following path: /signupOrganizer.

Parameters

<i>signup</i>	SignupAttempt object containing the information of the organizer requested when registering for the first time (name, email, password and organization).
---------------	--

Returns

Response object with the information of the organizer, in case of correct registration. If not, an empty response is returned.

Since

Sprint 2

7.35.3.7 attemptOrganizerUpdate()

```
Response es.deusto.server.server.Server.attemptOrganizerUpdate (
    SignupAttempt signup )
```

This method is used to update the information of an Organizer account.

It is invoked whenever a POST request is made to the following path: /updateOrganizer .

Parameters

<i>signup</i>	SignupAttempt object containing the information of the Organizers (name, email, password and organization). In this case it is also used for updating their information since its basic structure doesn't change.
---------------	---

Since

Sprint 2

Returns

Response object with the information of the organizer, in case of correct registration. If not, an empty response is returned

7.35.3.8 attemptUserDelete()

```
Response es.deusto.server.server.Server.attemptUserDelete (
    UserInfo userInfo )
```

This method is used of the delete method of regular Users.

It is invoked whenever a POST request is made to the following path: /delete .

Parameters

<i>userInfo</i>	UserInfo object containing user's email.
-----------------	--

Since

Sprint 2

Returns

Response string with the user's email, in case of user delete was successful. If not, an empty response is returned.

7.35.3.9 createEvent()

```
Response es.deusto.server.server.Server.createEvent (
    EventInfo eventInfo )
```

This method is used for the creation of new events from a given organizer.

It is invoked whenever a POST request is made to the following path: /createEvent.

Parameters

<i>eventInfo</i>	EventInfo object with the data of the event that will be created.
------------------	---

Returns

Response object containing the information of a EventInfo.

Since

Sprint 2

7.35.3.10 createPost()

```
Response es.deusto.server.server.Server.createPost (
    PostInfo postInfo )
```

This method is used for the creation of new Post from a given Event.

It is invoked whenever a POST request is made to the following path: /createEvent.

Parameters

<i>postInfo</i>	PostInfo object with the data of the event that will be created.
-----------------	--

Returns

Response object containing the information of a PostInfo.

Since

Sprint 2

7.35.3.11 getRecommendedEvents()

```
Response es.deusto.server.server.Server.getRecommendedEvents (
    SignupAttempt signupAttempt )
```

This method is used for recommend Event to users.

It is invoked whenever a POST request is made to the following path: /recomendation .

Parameters

<i>signupAttempt</i>	SignupAttempt object containing the interests and locations of the User.
----------------------	--

Returns

Response object with the events information

Since

Sprint 3

7.35.3.12 recoverPassword()

```
Response es.deusto.server.server.Server.recoverPassword (
    LoginAttempt userInfo )
```

This method is used for password recovery for users who can't remember their passwords.

It is invoked whenever a POST request is made to the following path: /passwordRecovery .

Parameters

<i>userInfo</i>	LoginAttempt object with the email of the user who needs a new password.
-----------------	--

Returns

Response object with the following message: "OK".

Since

Sprint 1

7.35.3.13 sayHello()

```
Response es.deusto.server.server.Server.sayHello ( )
```

method used for manual testing

Returns

Deprecated

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

- [src/main/java/es/deusto/server/server/Server.java](#)

7.36 es.deusto.junit.server.ServerTest Class Reference

This classes is used to create several tests to see if the server is working as intended.

Public Member Functions

- void [testNormalSignup](#) ()
- void [testNormalUpdate](#) ()
- void [testOrganizerSignup](#) ()
- void [testOrganizerLogin](#) ()
- void [testOrganizerUpdate](#) ()
- void [testCreateEvent](#) ()
- void [testCreatePost](#) ()

Static Public Member Functions

- static void [startUp](#) ()
- static junit.framework.Test [suite](#) ()
- static void [setUp](#) ()
- static void [tearDown](#) ()

7.36.1 Detailed Description

This classes is used to create several tests to see if the server is working as intended.

7.36.2 Member Function Documentation

7.36.2.1 [setUp\(\)](#)

```
static void es.deusto.junit.server.ServerTest.setUp ( ) [static]
```

7.36.2.2 [startUp\(\)](#)

```
static void es.deusto.junit.server.ServerTest.startUp ( ) [static]
```

7.36.2.3 [suite\(\)](#)

```
static junit.framework.Test es.deusto.junit.server.ServerTest.suite ( ) [static]
```

7.36.2.4 [tearDown\(\)](#)

```
static void es.deusto.junit.server.ServerTest.tearDown ( ) [static]
```

7.36.2.5 [testCreateEvent\(\)](#)

```
void es.deusto.junit.server.ServerTest.testCreateEvent ( )
```

7.36.2.6 testCreatePost()

```
void es.deusto.junit.server.ServerTest.testCreatePost ( )
```

7.36.2.7 testNormalSignup()

```
void es.deusto.junit.server.ServerTest.testNormalSignup ( )
```

7.36.2.8 testNormalUpdate()

```
void es.deusto.junit.server.ServerTest.testNormalUpdate ( )
```

7.36.2.9 testOrganizerLogin()

```
void es.deusto.junit.server.ServerTest.testOrganizerLogin ( )
```

7.36.2.10 testOrganizerSignup()

```
void es.deusto.junit.server.ServerTest.testOrganizerSignup ( )
```

7.36.2.11 testOrganizerUpdate()

```
void es.deusto.junit.server.ServerTest.testOrganizerUpdate ( )
```

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

- [src/test/java/es/deusto/junit/server/ServerTest.java](#)

7.37 es.deusto.serialization.SignupAttempt Class Reference

This class encapsulates all the information required when a user (Either normal User or Organizer) signs up.

Public Member Functions

- [SignupAttempt](#) ()
- [SignupAttempt](#) ([UserInfo](#) userInfo)
- String [getEmail](#) ()
- void [setEmail](#) (String email)
- String [getPassword](#) ()
- void [setPassword](#) (String password)
- String [getName](#) ()
- void [setName](#) (String name)
- String [getCity](#) ()
- void [setCity](#) (String city)
- String [getOrganization](#) ()
- void [setOrganization](#) (String organization)
- ArrayList< [TopicInfo](#) > [getInterests](#) ()
- void [setInterests](#) (ArrayList< [TopicInfo](#) > interests)
- ArrayList< [EventInfo](#) > [getSavedEvents](#) ()
- void [setSaveEvents](#) (ArrayList< [EventInfo](#) > savedEvents)
- [User](#) [buildUser](#) ()
 - Builds an instance of the business object User.*
- [Organizer](#) [buildOrganizer](#) ()
 - Builds an instance of the business object Organizer.*
- String [getCountry](#) ()
- void [setCountry](#) (String country)
- String [toString](#) ()

7.37.1 Detailed Description

This class encapsulates all the information required when a user (Either normal User or Organizer) signs up.

Since

Sprint 1

7.37.2 Constructor & Destructor Documentation

7.37.2.1 SignupAttempt() [1/2]

```
es.deusto.serialization.SignupAttempt.SignupAttempt ( )
```

7.37.2.2 SignupAttempt() [2/2]

```
es.deusto.serialization.SignupAttempt.SignupAttempt (
    UserInfo userInfo )
```


7.37.3 Member Function Documentation

7.37.3.1 buildOrganizer()

`Organizer` es.deusto.serialization.SignupAttempt.buildOrganizer ()

Builds an instance of the business object Organizer.

The main purpose of this function is prevent object repetition. It's only used in the server.

Returns

instance of Organizer.

7.37.3.2 buildUser()

`User` es.deusto.serialization.SignupAttempt.buildUser ()

Builds an instance of the business object User.

The main purpose of this function is prevent object repetition. It's only used in the server.

Returns

instance of es.deusto.server.data.User.java

7.37.3.3 getCity()

`String` es.deusto.serialization.SignupAttempt.getCity ()

7.37.3.4 getCountry()

`String` es.deusto.serialization.SignupAttempt.getCountry ()

7.37.3.5 getEmail()

`String` es.deusto.serialization.SignupAttempt.getEmail ()

7.37.3.6 getInterests()

```
ArrayList<TopicInfo> es.deusto.serialization.SignupAttempt.getInterests ( )
```

7.37.3.7 getName()

```
String es.deusto.serialization.SignupAttempt.getName ( )
```

7.37.3.8 getOrganization()

```
String es.deusto.serialization.SignupAttempt.getOrganization ( )
```

7.37.3.9 getPassword()

```
String es.deusto.serialization.SignupAttempt.getPassword ( )
```

7.37.3.10 getSavedEvents()

```
ArrayList<EventInfo> es.deusto.serialization.SignupAttempt.getSavedEvents ( )
```

7.37.3.11 setCity()

```
void es.deusto.serialization.SignupAttempt.setCity (
    String city )
```

7.37.3.12 setCountry()

```
void es.deusto.serialization.SignupAttempt.setCountry (
    String country )
```

7.37.3.13 setEmail()

```
void es.deusto.serialization.SignupAttempt.setEmail (
    String email )
```

7.37.3.14 setInterests()

```
void es.deusto.serialization.SignupAttempt.setInterests (
    ArrayList< TopicInfo > interests )
```

7.37.3.15 setName()

```
void es.deusto.serialization.SignupAttempt.setName (
    String name )
```

7.37.3.16 setOrganization()

```
void es.deusto.serialization.SignupAttempt.setOrganization (
    String organization )
```

7.37.3.17 setPassword()

```
void es.deusto.serialization.SignupAttempt.setPassword (
    String password )
```

7.37.3.18 setSaveEvents()

```
void es.deusto.serialization.SignupAttempt.setSaveEvents (
    ArrayList< EventInfo > savedEvents )
```

7.37.3.19 toString()

```
String es.deusto.serialization.SignupAttempt.toString ( )
```

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

- src/main/java/es/deusto/serialization/[SignupAttempt.java](#)

7.38 es.deusto.server.data.Topic Class Reference

This classes is the one that defines a topic and it is used to create one.

Public Member Functions

- [Topic](#) ()
- [Topic](#) ([TopicInfo](#) info)
- String [getName](#) ()
- void [setName](#) (String name)
- String [toString](#) ()

7.38.1 Detailed Description

This classes is the one that defines a topic and it is used to create one.

7.38.2 Constructor & Destructor Documentation

7.38.2.1 [Topic\(\)](#) [1/2]

```
es.deusto.server.data.Topic.Topic ( )
```

7.38.2.2 [Topic\(\)](#) [2/2]

```
es.deusto.server.data.Topic.Topic (
    TopicInfo info )
```

7.38.3 Member Function Documentation

7.38.3.1 [getName\(\)](#)

```
String es.deusto.server.data.Topic.getName ( )
```

7.38.3.2 setName()

```
void es.deusto.server.data.Topic.setName (
    String name )
```

7.38.3.3 toString()

```
String es.deusto.server.data.Topic.toString ( )
```

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

- [src/main/java/es/deusto/server/data/Topic.java](#)

7.39 es.deusto.server.dao.TopicDAO Class Reference

This classes is the one used to create a DAO a topic.

Public Member Functions

- [Topic](#) [getTopic](#) (String name)
- void [storeTopic](#) ([Topic](#) topic)
- void [updateTopic](#) ([Topic](#) topic)
- void [deleteTopic](#) (String name)

Protected Member Functions

- [TopicDAO](#) (PersistenceManager pm)
- PersistenceManager [getPersistenceManager](#) ()
- void [close](#) ()

7.39.1 Detailed Description

This classes is the one used to create a DAO a topic.

7.39.2 Constructor & Destructor Documentation

7.39.2.1 TopicDAO()

```
es.deusto.server.dao.TopicDAO.TopicDAO (
    PersistenceManager pm ) [protected]
```

7.39.3 Member Function Documentation

7.39.3.1 close()

```
void es.deusto.server.dao.TopicDAO.close ( ) [protected]
```

7.39.3.2 deleteTopic()

```
void es.deusto.server.dao.TopicDAO.deleteTopic (
    String name )
```

7.39.3.3 getPersistenceManager()

```
PersistenceManager es.deusto.server.dao.TopicDAO.getPersistenceManager ( ) [protected]
```

7.39.3.4 getTopic()

```
Topic es.deusto.server.dao.TopicDAO.getTopic (
    String name )
```

7.39.3.5 storeTopic()

```
void es.deusto.server.dao.TopicDAO.storeTopic (
    Topic topic )
```

7.39.3.6 updateTopic()

```
void es.deusto.server.dao.TopicDAO.updateTopic (
    Topic topic )
```

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

- src/main/java/es/deusto/server/dao/TopicDAO.java

7.40 es.deusto.serialization.TopicInfo Class Reference

DTO equivalent class of the Topic class from the server.

Public Member Functions

- TopicInfo ()
- TopicInfo (String name)
- TopicInfo (Topic c)
- String getName ()
- void setName (String name)
- String toString ()

Static Public Member Functions

- static Topic parseTopic (TopicInfo topicInfo)
Creates a Topic object from a TopicInfo one.

7.40.1 Detailed Description

DTO equivalent class of the Topic class from the server.

This class is used to transport all the relevant information of a topic through the network.

Since

Sprint 1

7.40.2 Constructor & Destructor Documentation

7.40.2.1 TopicInfo() [1/3]

```
es.deusto.serialization.TopicInfo.TopicInfo ( )
```

7.40.2.2 TopicInfo() [2/3]

```
es.deusto.serialization.TopicInfo.TopicInfo (
    String name )
```

7.40.2.3 TopicInfo() [3/3]

```
es.deusto.serialization.TopicInfo.TopicInfo (
    Topic c )
```

7.40.3 Member Function Documentation

7.40.3.1 getName()

```
String es.deusto.serialization.TopicInfo.getName ( )
```

7.40.3.2 parseTopic()

```
static Topic es.deusto.serialization.TopicInfo.parseTopic (
    TopicInfo topicInfo ) [static]
```

Creates a Topic object from a [TopicInfo](#) one.

Parameters

<i>topicInfo</i>	the source of the information
------------------	-------------------------------

Returns

an Object Topic

7.40.3.3 setName()

```
void es.deusto.serialization.TopicInfo.setName (
    String name )
```


7.40.3.4 toString()

```
String es.deusto.serialization.TopicInfo.toString ( )
```

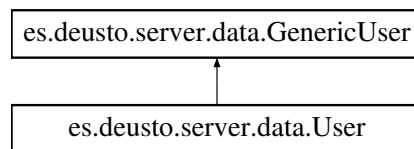
The documentation for this class was generated from the following file:

- [src/main/java/es/deusto/serialization/TopicInfo.java](#)

7.41 es.deusto.server.data.User Class Reference

This classes is the one that defines a normal user and it is used to create one.

Inheritance diagram for es.deusto.server.data.User:



Public Member Functions

- String [getCity](#) ()
- String [getCountry](#) ()
- void [setCountry](#) (String country)
- void [setCity](#) (String city)
- ArrayList< [Event](#) > [getSavedEvents](#) ()
- void [setSavedEvents](#) (ArrayList< [Event](#) > savedEvents)
- void [addEvent](#) ([Event](#) event)
- ArrayList< [Topic](#) > [getInterests](#) ()
- void [setInterests](#) (ArrayList< [Topic](#) > topics)
- void [addInterest](#) ([Topic](#) topic)
- String [toString](#) ()

7.41.1 Detailed Description

This classes is the one that defines a normal user and it is used to create one.

7.41.2 Member Function Documentation

7.41.2.1 addEvent()

```
void es.deusto.server.data.User.addEvent (
    Event event )
```

7.41.2.2 addInterest()

```
void es.deusto.server.data.User.addInterest (
    Topic topic )
```

7.41.2.3 getCity()

```
String es.deusto.server.data.User.getCity ( )
```

7.41.2.4 getCountry()

```
String es.deusto.server.data.User.getCountry ( )
```

7.41.2.5 getInterests()

```
ArrayList<Topic> es.deusto.server.data.User.getInterests ( )
```

7.41.2.6 getSavedEvents()

```
ArrayList<Event> es.deusto.server.data.User.getSavedEvents ( )
```

7.41.2.7 setCity()

```
void es.deusto.server.data.User.setCity (
    String city )
```

7.41.2.8 setCountry()

```
void es.deusto.server.data.User.setCountry (
    String country )
```

7.41.2.9 setInterests()

```
void es.deusto.server.data.User.setInterests (
    ArrayList< Topic > topics )
```

7.41.2.10 setSavedEvents()

```
void es.deusto.server.data.User.setSavedEvents (
    ArrayList< Event > savedEvents )
```

7.41.2.11 toString()

```
String es.deusto.server.data.User.toString ( )
```

Reimplemented from [es.deusto.server.data.GenericUser](#).

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

- [src/main/java/es/deusto/server/data/User.java](#)

7.42 es.deusto.server.dao.UserDAO Class Reference

This classes is the one used to create an user DAO.

Public Member Functions

- [User](#) [getUser](#) (String email)
- void [storeUser](#) ([User](#) user)
- void [updateUser](#) ([User](#) user)
- void [deleteUser](#) (String email)

Protected Member Functions

- [UserDAO](#) (PersistenceManager pm)
- PersistenceManager [getPersistenceManager](#) ()
- void [close](#) ()

7.42.1 Detailed Description

This classes is the one used to create an user DAO.

7.42.2 Constructor & Destructor Documentation

7.42.2.1 UserDAO()

```
es.deusto.server.dao.UserDAO.UserDAO (
    PersistenceManager pm ) [protected]
```

7.42.3 Member Function Documentation

7.42.3.1 close()

```
void es.deusto.server.dao.UserDAO.close ( ) [protected]
```

7.42.3.2 deleteUser()

```
void es.deusto.server.dao.UserDAO.deleteUser (
    String email )
```

7.42.3.3 getPersistenceManager()

```
PersistenceManager es.deusto.server.dao.UserDAO.getPersistenceManager ( ) [protected]
```

7.42.3.4 getUser()

```
User es.deusto.server.dao.UserDAO.getUser (
    String email )
```

7.42.3.5 storeUser()

```
void es.deusto.server.dao.UserDAO.storeUser (
    User user )
```

7.42.3.6 updateUser()

```
void es.deusto.server.dao.UserDAO.updateUser (
    User user )
```

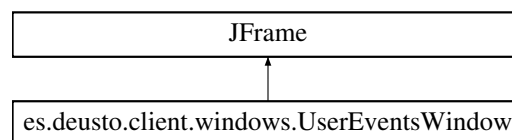
The documentation for this class was generated from the following file:

- [src/main/java/es/deusto/server/dao/UserDAO.java](#)

7.43 es.deusto.client.windows.UserEventsWindow Class Reference

This class is the one used to create a window that allows a user to scroll, search and select from a list of events he previously has chosen from.

Inheritance diagram for es.deusto.client.windows.UserEventsWindow:



Public Member Functions

- [UserEventsWindow](#) ([Controller](#) controller)

7.43.1 Detailed Description

This class is the one used to create a window that allows a user to scroll, search and select from a list of events he previously has chosen from.

A personal event list.

7.43.2 Constructor & Destructor Documentation

7.43.2.1 UserEventsWindow()

```
es.deusto.client.windows.UserEventsWindow.UserEventsWindow (
    Controller controller )
```

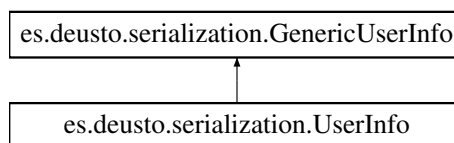
The documentation for this class was generated from the following file:

- [src/main/java/es/deusto/client/windows/UserEventsWindow.java](#)

7.44 es.deusto.serialization.UserInfo Class Reference

DTO equivalent class of the User class from the server.

Inheritance diagram for es.deusto.serialization.UserInfo:



Public Member Functions

- [UserInfo](#) (String name, String email, String city, String country)
- [UserInfo](#) ()
- [UserInfo](#) ([User](#) user)
- String [getName](#) ()
- void [setName](#) (String name)
- String [getEmail](#) ()
- void [setEmail](#) (String email)
- String [getCity](#) ()
- void [setCity](#) (String city)
- ArrayList< [EventInfo](#) > [getSavedEvents](#) ()
- void [setSavedEvents](#) (ArrayList< [EventInfo](#) > savedEvents)
- ArrayList< [TopicInfo](#) > [getInterests](#) ()
- void [setInterests](#) (ArrayList< [TopicInfo](#) > interests)
- void [addInterest](#) ([TopicInfo](#) interest)
- String [getCountry](#) ()
- void [setCountry](#) (String country)

7.44.1 Detailed Description

DTO equivalent class of the User class from the server.

This class is used to transport all the relevant information of a user through the network.

Since

Sprint 1

7.44.2 Constructor & Destructor Documentation

7.44.2.1 UserInfo() [1/3]

```
es.deusto.serialization.UserInfo.UserInfo (
    String name,
    String email,
    String city,
    String country )
```

7.44.2.2 UserInfo() [2/3]

```
es.deusto.serialization.UserInfo.UserInfo ( )
```

7.44.2.3 UserInfo() [3/3]

```
es.deusto.serialization.UserInfo.UserInfo (
    User user )
```

7.44.3 Member Function Documentation

7.44.3.1 addInterest()

```
void es.deusto.serialization.UserInfo.addInterest (
    TopicInfo interest )
```

7.44.3.2 getCity()

```
String es.deusto.serialization.UserInfo.getCity ( )
```

7.44.3.3 getCountry()

```
String es.deusto.serialization.UserInfo.getCountry ( )
```

7.44.3.4 getEmail()

```
String es.deusto.serialization.UserInfo.getEmail ( )
```

7.44.3.5 getInterests()

```
ArrayList<TopicInfo> es.deusto.serialization.UserInfo.getInterests ( )
```

7.44.3.6 getName()

```
String es.deusto.serialization.UserInfo.getName ( )
```

7.44.3.7 getSavedEvents()

```
ArrayList<EventInfo> es.deusto.serialization.UserInfo.getSavedEvents ( )
```

7.44.3.8 setCity()

```
void es.deusto.serialization.UserInfo.setCity (
    String city )
```

7.44.3.9 setCountry()

```
void es.deusto.serialization.UserInfo.setCountry (
    String country )
```

7.44.3.10 setEmail()

```
void es.deusto.serialization.UserInfo.setEmail (
    String email )
```


7.44.3.11 setInterests()

```
void es.deusto.serialization.UserInfo.setInterests (
    ArrayList< TopicInfo > interests )
```

7.44.3.12 setName()

```
void es.deusto.serialization.UserInfo.setName (
    String name )
```

7.44.3.13 setSavedEvents()

```
void es.deusto.serialization.UserInfo.setSavedEvents (
    ArrayList< EventInfo > savedEvents )
```

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

- [src/main/java/es/deusto/serialization/UserInfo.java](#)

Chapter 8

File Documentation

8.1 src/main/java/es/deusto/client/controller/Controller.java File Reference

Classes

- class [es.deusto.client.controller.Controller](#)
This class is the one carrying out all the communication with the server.

Packages

- package [es.deusto.client.controller](#)

8.2 src/main/java/es/deusto/client/main/Main.java File Reference

Classes

- class [es.deusto.client.main.Main](#)
This class is the one that starts the application.

Packages

- package [es.deusto.client.main](#)

8.3 src/main/java/es/deusto/client/windows/ChangePass.java File Reference

Classes

- class [es.deusto.client.windows.ChangePass](#)
This class is the one used to create a window to change the user password.

Packages

- package [es.deusto.client.windows](#)

8.4 src/main/java/es/deusto/client/windows/CodeVerification.java File Reference

Classes

- class [es.deusto.client.windows.CodeVerification](#)

Packages

- package [es.deusto.client.windows](#)

8.5 src/main/java/es/deusto/client/windows/CreateEvent.java File Reference

Classes

- class [es.deusto.client.windows.CreateEvent](#)

This class is the one used to create a window that it allows the user to create a new event.

Packages

- package [es.deusto.client.windows](#)

8.6 src/main/java/es/deusto/client/windows/CreateOrganizerWindow.java File Reference

Classes

- class [es.deusto.client.windows.CreateOrganizerWindow](#)

This class is the one used to create a window that allows to create a new organizer.

Packages

- package [es.deusto.client.windows](#)

8.7 src/main/java/es/deusto/client/windows/CreateUserWindow.java File Reference

Classes

- class [es.deusto.client.windows.CreateUserWindow](#)
This class is the one used to create a window that allows to create a new user.

Packages

- package [es.deusto.client.windows](#)

8.8 src/main/java/es/deusto/client/windows/EventListItem.java File Reference

Classes

- class [es.deusto.client.windows.EventListItem](#)
This class is the one used to create a window that allows the user to have an event list.

Packages

- package [es.deusto.client.windows](#)

8.9 src/main/java/es/deusto/client/windows/EventOrganizerWindow.java File Reference

Classes

- class [es.deusto.client.windows.EventOrganizerWindow](#)
This class is the one used to create a window that allows to an organizer to change or administrate one of his events.

Packages

- package [es.deusto.client.windows](#)

8.10 src/main/java/es/deusto/client/windows/EventWindow.java File Reference

Classes

- class [es.deusto.client.windows.EventWindow](#)
This class is the one used to create a window that allows to display all the information of one event.

Packages

- package [es.deusto.client.windows](#)

8.11 src/main/java/es/deusto/client/windows/ForgottenPassword.java File Reference

Classes

- class [es.deusto.client.windows.ForgottenPassword](#)
This class is the one used to create a window that allows a user to log in when he forgets the password.

Packages

- package [es.deusto.client.windows](#)

8.12 src/main/java/es/deusto/client/windows/LanguageManager.java File Reference

Classes

- class [es.deusto.client.windows.LanguageManager](#)
This class is used to manage the different languages supported by the application.

Packages

- package [es.deusto.client.windows](#)

8.13 src/main/java/es/deusto/client/windows/LogInWindow.java File Reference

Classes

- class [es.deusto.client.windows.LogInWindow](#)
This class is the one used to create a window that allows a user/organizer to log in to the application.

Packages

- package [es.deusto.client.windows](#)

8.14 src/main/java/es/deusto/client/windows/OrganizerEvents.java File Reference

Classes

- class [es.deusto.client.windows.OrganizerEvents](#)

This class is the one used to create a window that allows an organizer to have all of his events.

Packages

- package [es.deusto.client.windows](#)

8.15 src/main/java/es/deusto/client/windows/OrganizerHome.java File Reference

Classes

- class [es.deusto.client.windows.OrganizerHome](#)

This class is the one used to create a window that allows an organizer to have a home window where he can access to all of the functionalities of the application.

Packages

- package [es.deusto.client.windows](#)

8.16 src/main/java/es/deusto/client/windows/PostCreationWindow.java File Reference

Classes

- class [es.deusto.client.windows.PostCreationWindow](#)

This class is the one used to create a window that allows an organizer to have like a double check of the event he has just created.

Packages

- package [es.deusto.client.windows](#)

8.17 src/main/java/es/deusto/client/windows/Profile.java File Reference

Classes

- class [es.deusto.client.windows.Profile](#)

This class is the one used to create a window that allows a user/organizer to have a profile with all of his information.

Packages

- package [es.deusto.client.windows](#)

8.18 src/main/java/es/deusto/client/windows/SavedEventsWindow.java File Reference

Classes

- class [es.deusto.client.windows.SavedEventsWindow](#)

This class is the one used to create a window that allows to scroll, search and choose from all of the events stored in the database.

Packages

- package [es.deusto.client.windows](#)

8.19 src/main/java/es/deusto/client/windows/UserEventsWindow.java File Reference

Classes

- class [es.deusto.client.windows.UserEventsWindow](#)

This class is the one used to create a window that allows a user to scroll, search and select from a list of events he previously has chosen from.

Packages

- package [es.deusto.client.windows](#)

8.20 src/main/java/es/deusto/serialization/EventInfo.java File Reference

Classes

- class [es.deusto.serialization.EventInfo](#)

DTO equivalent class of the Event class from the server.

Packages

- package [es.deusto.serialization](#)

8.21 src/main/java/es/deusto/serialization/GenericUserInfo.java File Reference

Classes

- class [es.deusto.serialization.GenericUserInfo](#)

This class works as a Tag for the Classes related to the User and Organizer.

Packages

- package [es.deusto.serialization](#)

8.22 src/main/java/es/deusto/serialization/LoginAttempt.java File Reference

Classes

- class [es.deusto.serialization.LoginAttempt](#)

This class encapsulates all the information required when a user (Either normal User or Organizer) logs in.

Packages

- package [es.deusto.serialization](#)

8.23 src/main/java/es/deusto/serialization/OrganizerInfo.java File Reference

Classes

- class [es.deusto.serialization.OrganizerInfo](#)

DTO equivalent class of the Organizer class from the server.

Packages

- package [es.deusto.serialization](#)

8.24 src/main/java/es/deusto/serialization/PostInfo.java File Reference

Classes

- class [es.deusto.serialization.PostInfo](#)

DTO equivalent class of the Post class from the server.

Packages

- package [es.deusto.serialization](#)

8.25 src/main/java/es/deusto/serialization/SignupAttempt.java File Reference

Classes

- class [es.deusto.serialization.SignupAttempt](#)
This class encapsulates all the information required when a user (Either normal User or Organizer) signs up.

Packages

- package [es.deusto.serialization](#)

8.26 src/main/java/es/deusto/serialization/TopicInfo.java File Reference

Classes

- class [es.deusto.serialization.TopicInfo](#)
DTO equivalent class of the Topic class from the server.

Packages

- package [es.deusto.serialization](#)

8.27 src/main/java/es/deusto/serialization/UserInfo.java File Reference

Classes

- class [es.deusto.serialization.UserInfo](#)
DTO equivalent class of the User class from the server.

Packages

- package [es.deusto.serialization](#)

8.28 src/main/java/es/deusto/server/dao/DAOFactory.java File Reference

Classes

- class [es.deusto.server.dao.DAOFactory](#)
This class is used to create a DAO factory that is used by all of the other DAO classes, to create them or to close them.

Packages

- package [es.deusto.server.dao](#)

8.29 src/main/java/es/deusto/server/dao/EventDAO.java File Reference

Classes

- class [es.deusto.server.dao.EventDAO](#)
This classes is the one used to create an event DAO.

Packages

- package [es.deusto.server.dao](#)

8.30 src/main/java/es/deusto/server/dao/OrganizerDAO.java File Reference

Classes

- class [es.deusto.server.dao.OrganizerDAO](#)
This classes is the one used to create an organizer DAO.

Packages

- package [es.deusto.server.dao](#)

8.31 src/main/java/es/deusto/server/dao/PostDAO.java File Reference

Classes

- class [es.deusto.server.dao.PostDAO](#)
This classes is the one used to create a DAO for a post.

Packages

- package [es.deusto.server.dao](#)

8.32 src/main/java/es/deusto/server/dao/TopicDAO.java File Reference

Classes

- class [es.deusto.server.dao.TopicDAO](#)
This classes is the one used to create a DAO a topic.

Packages

- package [es.deusto.server.dao](#)

8.33 src/main/java/es/deusto/server/dao/UserDAO.java File Reference

Classes

- class [es.deusto.server.dao.UserDAO](#)
This classes is the one used to create an user DAO.

Packages

- package [es.deusto.server.dao](#)

8.34 src/main/java/es/deusto/server/data/Event.java File Reference

Classes

- class [es.deusto.server.data.Event](#)
This classes is the one that defines an event and it is used to create one.

Packages

- package [es.deusto.server.data](#)

8.35 src/main/java/es/deusto/server/data/GenericUser.java File Reference

Classes

- class [es.deusto.server.data.GenericUser](#)
This classes is the father class for normal user and organizer.

Packages

- package [es.deusto.server.data](#)

8.36 src/main/java/es/deusto/server/data/Organizer.java File Reference

Classes

- class [es.deusto.server.data.Organizer](#)
This classes is the one that defines an organizer and it is used to create one.

Packages

- package [es.deusto.server.data](#)

8.37 src/main/java/es/deusto/server/data/Post.java File Reference

Classes

- class [es.deusto.server.data.Post](#)

This classes is the one that defines a post and it is used to create one.

Packages

- package [es.deusto.server.data](#)

8.38 src/main/java/es/deusto/server/data/Topic.java File Reference

Classes

- class [es.deusto.server.data.Topic](#)

This classes is the one that defines a topic and it is used to create one.

Packages

- package [es.deusto.server.data](#)

8.39 src/main/java/es/deusto/server/data/User.java File Reference

Classes

- class [es.deusto.server.data.User](#)

This classes is the one that defines a normal user and it is used to create one.

Packages

- package [es.deusto.server.data](#)

8.40 src/main/java/es/deusto/server/server/AppService.java File Reference

Classes

- class [es.deusto.server.server.AppService](#)

Packages

- package [es.deusto.server.server](#)

8.41 src/main/java/es/deusto/server/server/Server.java File Reference

Classes

- class [es.deusto.server.server.Server](#)
This class is the one that receives the requests from the client.

Packages

- package [es.deusto.server.server](#)

8.42 src/test/java/es/deusto/junit/client/ControllerTest.java File Reference

Classes

- class [es.deusto.junit.client.ControllerTest](#)
This classes is the one that tests the controller component to see if it is working as intended.

Packages

- package [es.deusto.junit.client](#)

8.43 src/test/java/es/deusto/junit/serialization/SerializationTest.java File Reference

Classes

- class [es.deusto.junit.serialization.SerializationTest](#)
This classes is the one that tests funcionalities such as creating an event, creating a user, orginzer, crea ting posts and topics.

Packages

- package [es.deusto.junit.serialization](#)

8.44 src/test/java/es/deusto/junit/server/ServerTest.java File Reference

Classes

- class [es.deusto.junit.server.ServerTest](#)
This classes is used to create several tests to see if the server is working as intended.

Packages

- package [es.deusto.junit.server](#)

Index

- actionPerformed
 - es.deusto.client.windows.CreateEvent, 39
 - es.deusto.client.windows.CreateOrganizerWindow, 40
 - es.deusto.client.windows.CreateUserWindow, 41
- addCreatedEvent
 - es.deusto.server.data.Organizer, 68
- addEvent
 - es.deusto.server.data.User, 105
- addInterest
 - es.deusto.serialization.UserInfo, 111
 - es.deusto.server.data.User, 106
- addInterested
 - es.deusto.serialization.EventInfo, 52
 - es.deusto.server.data.Event, 45
- AppService
 - es.deusto.server.server.AppService, 20
- attemptNormalLogin
 - es.deusto.client.controller.Controller, 29, 30
 - es.deusto.server.server.AppService, 20
 - es.deusto.server.server.Server, 87
- attemptNormalLoginOrganizer
 - es.deusto.client.controller.Controller, 31
- attemptNormalSignup
 - es.deusto.client.controller.Controller, 31
 - es.deusto.server.server.AppService, 20
 - es.deusto.server.server.Server, 87
- attemptNormalUpdate
 - es.deusto.client.controller.Controller, 32
 - es.deusto.server.server.AppService, 21
 - es.deusto.server.server.Server, 88
- attemptOrganizerDelete
 - es.deusto.client.controller.Controller, 32
 - es.deusto.server.server.Server, 88
- attemptOrganizerLogin
 - es.deusto.server.server.AppService, 21
 - es.deusto.server.server.Server, 89
- attemptOrganizerSignup
 - es.deusto.client.controller.Controller, 32
 - es.deusto.server.server.AppService, 22
 - es.deusto.server.server.Server, 89
- attemptOrganizerUpdate
 - es.deusto.client.controller.Controller, 33
 - es.deusto.server.server.AppService, 22
 - es.deusto.server.server.Server, 90
- attemptUserDelete
 - es.deusto.client.controller.Controller, 33
 - es.deusto.server.server.Server, 90
- bCreate
 - es.deusto.client.windows.LogInWindow, 66
- beforeClass
 - es.deusto.junit.serialization.SerializationTest, 84
- buildOrganizer
 - es.deusto.serialization.SignupAttempt, 97
- buildUser
 - es.deusto.serialization.SignupAttempt, 97
- cBoxAdmin
 - es.deusto.client.windows.LogInWindow, 66
- ChangePass
 - es.deusto.client.windows.ChangePass, 27
- close
 - es.deusto.server.dao.EventDAO, 49
 - es.deusto.server.dao.OrganizerDAO, 69
 - es.deusto.server.dao.PostDAO, 78
 - es.deusto.server.dao.TopicDAO, 102
 - es.deusto.server.dao.UserDAO, 108
- closeDAO
 - es.deusto.server.dao.DAOFactory, 42, 43
- CodeVerification
 - es.deusto.client.windows.CodeVerification, 27
- configure
 - es.deusto.junit.client.ControllerTest, 37
- Controller
 - es.deusto.client.controller.Controller, 29
- CreateEvent
 - es.deusto.client.windows.CreateEvent, 39
- createEvent
 - es.deusto.client.controller.Controller, 34
 - es.deusto.server.server.AppService, 23
 - es.deusto.server.server.Server, 91
- createEventDAO
 - es.deusto.server.dao.DAOFactory, 43
- createOrganizerDAO
 - es.deusto.server.dao.DAOFactory, 43
- CreateOrganizerWindow
 - es.deusto.client.windows.CreateOrganizerWindow, 40
- createPost
 - es.deusto.client.controller.Controller, 34
 - es.deusto.server.server.AppService, 23
 - es.deusto.server.server.Server, 91
- createPostDAO
 - es.deusto.server.dao.DAOFactory, 43
- createTopicDAO
 - es.deusto.server.dao.DAOFactory, 43
- createUserDAO
 - es.deusto.server.dao.DAOFactory, 43
- CreateUserWindow

- es.deusto.client.windows.CreateUserWindow, 41
- deleteEvent
 - es.deusto.server.dao.EventDAO, 49
- deleteOrganizer
 - es.deusto.server.dao.OrganizerDAO, 69
 - es.deusto.server.server.AppService, 24
- deletePost
 - es.deusto.server.dao.PostDAO, 78
- deleteTopic
 - es.deusto.server.dao.TopicDAO, 102
- deleteUser
 - es.deusto.server.dao.UserDAO, 108
 - es.deusto.server.server.AppService, 24
- es, 13
- es.deusto, 13
- es.deusto.client, 13
- es.deusto.client.controller, 13
- es.deusto.client.controller.Controller, 28
 - attemptNormalLogin, 29, 30
 - attemptNormalLoginOrganizer, 31
 - attemptNormalSignup, 31
 - attemptNormalUpdate, 32
 - attemptOrganizerDelete, 32
 - attemptOrganizerSignup, 32
 - attemptOrganizerUpdate, 33
 - attemptUserDelete, 33
 - Controller, 29
 - createEvent, 34
 - createPost, 34
 - getLanguageManager, 35
 - getOrganize, 35
 - getRecommendations, 35
 - getUser, 35
 - setOrganizer, 36
 - setUser, 36
- es.deusto.client.main, 14
- es.deusto.client.main.Main, 66
 - main, 66
- es.deusto.client.windows, 14
- es.deusto.client.windows.ChangePass, 26
 - ChangePass, 27
- es.deusto.client.windows.CodeVerification, 27
 - CodeVerification, 27
- es.deusto.client.windows.CreateEvent, 39
 - actionPerformed, 39
 - CreateEvent, 39
- es.deusto.client.windows.CreateOrganizerWindow, 40
 - actionPerformed, 40
 - CreateOrganizerWindow, 40
- es.deusto.client.windows.CreateUserWindow, 41
 - actionPerformed, 41
 - CreateUserWindow, 41
- es.deusto.client.windows.EventListItem, 55
 - EventListItem, 56
- es.deusto.client.windows.EventOrganizerWindow, 56
 - EventOrganizerWindow, 57
- es.deusto.client.windows.EventWindow, 57
 - EventWindow, 57
- es.deusto.client.windows.ForgottenPassword, 58
 - ForgottenPassword, 58
- es.deusto.client.windows.LanguageManager, 61
 - getLanguage, 62
 - getString, 62
 - LanguageManager, 61
 - setLanguage, 62
- es.deusto.client.windows.LogInWindow, 65
 - bCreate, 66
 - cBoxAdmin, 66
 - id, 66
 - LogInWindow, 65
- es.deusto.client.windows.OrganizerEvents, 70
 - OrganizerEvents, 71
- es.deusto.client.windows.OrganizerHome, 71
 - OrganizerHome, 72
- es.deusto.client.windows.PostCreationWindow, 77
 - PostCreationWindow, 77
- es.deusto.client.windows.Profile, 82
 - Profile, 82
- es.deusto.client.windows.SavedEventsWindow, 82
 - SavedEventsWindow, 83
- es.deusto.client.windows.UserEventsWindow, 109
 - UserEventsWindow, 109
- es.deusto.junit, 15
- es.deusto.junit.client, 15
- es.deusto.junit.client.ControllerTest, 36
 - configure, 37
 - rule, 38
 - setUpChild, 37
 - startUp, 37
 - suite, 37
 - tearDownChild, 37
 - testNormalLogin, 38
 - testNormalSignupAndDelete, 38
 - testOrganizerLogin, 38
 - testOrganizerSignupAndDelete, 38
 - testUserUpdate, 38
- es.deusto.junit.serialization, 15
- es.deusto.junit.serialization.SerializationTest, 83
 - beforeClass, 84
 - startUp, 84
 - suite, 84
 - testEventInfoFromEvent, 84
 - testEventInfoSetters, 84
 - testOrganizerInfoFromOrganizer, 84
 - testOrganizerInfoSetters, 84
 - testPostInfoFromPost, 85
 - testPostInfoSetters, 85
 - testSignupAttemptOrganizer, 85
 - testSignupAttemptUser, 85
 - testUserInfoFromUser, 85
 - testUserInfoSetters, 85
 - testUserInfoStringConstructor, 85
- es.deusto.junit.server, 15
- es.deusto.junit.server.ServerTest, 93
 - setUp, 94

- startUp, 94
- suite, 94
- tearDown, 94
- testCreateEvent, 94
- testCreatePost, 94
- testNormalSignup, 95
- testNormalUpdate, 95
- testOrganizerLogin, 95
- testOrganizerSignup, 95
- testOrganizerUpdate, 95
- es.deusto.serialization, 15
- es.deusto.serialization.EventInfo, 51
 - addInterested, 52
 - EventInfo, 52
 - getCity, 53
 - getCountry, 53
 - getDate, 53
 - getDescription, 53
 - getInterested, 53
 - getName, 53
 - getOrganizerEmail, 53
 - getPosts, 53
 - getTopic, 54
 - reduceInterested, 54
 - setCity, 54
 - setCountry, 54
 - setDate, 54
 - setDescription, 54
 - setInterested, 54
 - setName, 55
 - setOrganizerEmail, 55
 - setPosts, 55
 - setTopic, 55
- es.deusto.serialization.GenericUserInfo, 60
- es.deusto.serialization.LoginAttempt, 62
 - getEmail, 64
 - getPassword, 64
 - isOrganizer, 64
 - LoginAttempt, 63
 - setEmail, 64
 - setOrganizer, 64
 - setPassword, 64
 - toString, 64
- es.deusto.serialization.OrganizerInfo, 72
 - getCreatedEvents, 73
 - getEmail, 73
 - getName, 73
 - getOrganization, 73
 - OrganizerInfo, 73
 - setCreatedEvents, 73
 - setEmail, 74
 - setName, 74
 - setOrganization, 74
- es.deusto.serialization.PostInfo, 79
 - getDate, 80
 - getDescription, 80
 - getEventName, 80
 - getOrganizerEmail, 80
 - getTitle, 80
 - PostInfo, 80
 - setDate, 81
 - setDescription, 81
 - setEventName, 81
 - setOrganizerEmail, 81
 - setTitle, 81
- es.deusto.serialization.SignupAttempt, 95
 - buildOrganizer, 97
 - buildUser, 97
 - getCity, 97
 - getCountry, 97
 - getEmail, 97
 - getInterests, 97
 - getName, 98
 - getOrganization, 98
 - getPassword, 98
 - getSavedEvents, 98
 - setCity, 98
 - setCountry, 98
 - setEmail, 98
 - setInterests, 99
 - setName, 99
 - setOrganization, 99
 - setPassword, 99
 - setSaveEvents, 99
 - SignupAttempt, 96
 - toString, 99
- es.deusto.serialization.TopicInfo, 103
 - getName, 104
 - parseTopic, 104
 - setName, 104
 - TopicInfo, 103, 104
 - toString, 104
- es.deusto.serialization.UserInfo, 110
 - addInterest, 111
 - getCity, 111
 - getCountry, 111
 - getEmail, 111
 - getInterests, 112
 - getName, 112
 - getSavedEvents, 112
 - setCity, 112
 - setCountry, 112
 - setEmail, 112
 - setInterests, 112
 - setName, 113
 - setSavedEvents, 113
 - UserInfo, 111
- es.deusto.server, 16
- es.deusto.server.dao, 16
- es.deusto.server.dao.DAOFactory, 42
 - closeDAO, 42, 43
 - createEventDAO, 43
 - createOrganizerDAO, 43
 - createPostDAO, 43
 - createTopicDAO, 43
 - createUserDAO, 43

- getInstance, 44
- es.deusto.server.dao.EventDAO, 48
 - close, 49
 - deleteEvent, 49
 - EventDAO, 49
 - getEvents, 49
 - getEventsByOrganizer, 50
 - getEventsbyUser, 50
 - getPersistenceManager, 50
 - storeEvent, 51
 - updateEvent, 51
- es.deusto.server.dao.OrganizerDAO, 69
 - close, 69
 - deleteOrganizer, 69
 - getOrganizer, 70
 - getPersistenceManager, 70
 - OrganizerDAO, 69
 - storeOrganizer, 70
 - updateOrganizer, 70
- es.deusto.server.dao.PostDAO, 77
 - close, 78
 - deletePost, 78
 - getPersistenceManager, 78
 - getPostsByEvent, 78
 - PostDAO, 78
 - storePost, 79
 - updatePost, 79
- es.deusto.server.dao.TopicDAO, 101
 - close, 102
 - deleteTopic, 102
 - getPersistenceManager, 102
 - getTopic, 102
 - storeTopic, 102
 - TopicDAO, 101
 - updateTopic, 102
- es.deusto.server.dao.UserDAO, 107
 - close, 108
 - deleteUser, 108
 - getPersistenceManager, 108
 - getUser, 108
 - storeUser, 108
 - updateUser, 109
 - UserDAO, 108
- es.deusto.server.data, 16
- es.deusto.server.data.Event, 44
 - addInterested, 45
 - Event, 45
 - getCity, 45
 - getCountry, 45
 - getDate, 45
 - getDescription, 45
 - getInterested, 46
 - getName, 46
 - getOrganizer, 46
 - getPosts, 46
 - getTopic, 46
 - reduceInterested, 46
 - setCity, 46
 - setCountry, 46
 - setDate, 47
 - setDescription, 47
 - setInterested, 47
 - setName, 47
 - setOrganizer, 47
 - setPosts, 47
 - setTopic, 47
 - toString, 48
- es.deusto.server.data.GenericUser, 58
 - getEmail, 59
 - getName, 59
 - getPassword, 59
 - setEmail, 59
 - setName, 59
 - setPassword, 60
 - toString, 60
- es.deusto.server.data.Organizer, 67
 - addCreatedEvent, 68
 - getCreatedEvents, 68
 - getOrganization, 68
 - Organizer, 67
 - setCreatedEvents, 68
 - setOrganization, 68
 - toString, 68
- es.deusto.server.data.Post, 74
 - getDate, 75
 - getDescription, 75
 - getEventName, 75
 - getOrganizerEmail, 75
 - getTitle, 76
 - Post, 75
 - setDate, 76
 - setDescription, 76
 - setEventName, 76
 - setOrganizerEmail, 76
 - setTitle, 76
- es.deusto.server.data.Topic, 100
 - getName, 100
 - setName, 100
 - Topic, 100
 - toString, 101
- es.deusto.server.data.User, 105
 - addEvent, 105
 - addInterest, 106
 - getCity, 106
 - getCountry, 106
 - getInterests, 106
 - getSavedEvents, 106
 - setCity, 106
 - setCountry, 106
 - setInterests, 107
 - setSavedEvents, 107
 - toString, 107
- es.deusto.server.server, 17
- es.deusto.server.server.AppService, 19
 - AppService, 20
 - attemptNormalLogin, 20

- attemptNormalSignup, 20
- attemptNormalUpdate, 21
- attemptOrganizerLogin, 21
- attemptOrganizerSignup, 22
- attemptOrganizerUpdate, 22
- createEvent, 23
- createPost, 23
- deleteOrganizer, 24
- deleteUser, 24
- getRecommendedEvents, 25
- hello, 25
- recoverPassword, 25
- updateUser, 26
- es.deusto.server.server.Server, 86
 - attemptNormalLogin, 87
 - attemptNormalSignup, 87
 - attemptNormalUpdate, 88
 - attemptOrganizerDelete, 88
 - attemptOrganizerLogin, 89
 - attemptOrganizerSignup, 89
 - attemptOrganizerUpdate, 90
 - attemptUserDelete, 90
 - createEvent, 91
 - createPost, 91
 - getRecommendedEvents, 92
 - recoverPassword, 92
 - sayHello, 93
 - Server, 87
- Event
 - es.deusto.server.data.Event, 45
- EventDAO
 - es.deusto.server.dao.EventDAO, 49
- EventInfo
 - es.deusto.serialization.EventInfo, 52
- EventListItem
 - es.deusto.client.windows.EventListItem, 56
- EventOrganizerWindow
 - es.deusto.client.windows.EventOrganizerWindow, 57
- EventWindow
 - es.deusto.client.windows.EventWindow, 57
- ForgottenPassword
 - es.deusto.client.windows.ForgottenPassword, 58
- getCity
 - es.deusto.serialization.EventInfo, 53
 - es.deusto.serialization.SignupAttempt, 97
 - es.deusto.serialization.UserInfo, 111
 - es.deusto.server.data.Event, 45
 - es.deusto.server.data.User, 106
- getCountry
 - es.deusto.serialization.EventInfo, 53
 - es.deusto.serialization.SignupAttempt, 97
 - es.deusto.serialization.UserInfo, 111
 - es.deusto.server.data.Event, 45
 - es.deusto.server.data.User, 106
- getCreatedEvents
 - es.deusto.serialization.OrganizerInfo, 73
 - es.deusto.server.data.Organizer, 68
- getDate
 - es.deusto.serialization.EventInfo, 53
 - es.deusto.serialization.PostInfo, 80
 - es.deusto.server.data.Event, 45
 - es.deusto.server.data.Post, 75
- getDescription
 - es.deusto.serialization.EventInfo, 53
 - es.deusto.serialization.PostInfo, 80
 - es.deusto.server.data.Event, 45
 - es.deusto.server.data.Post, 75
- getEmail
 - es.deusto.serialization.LoginAttempt, 64
 - es.deusto.serialization.OrganizerInfo, 73
 - es.deusto.serialization.SignupAttempt, 97
 - es.deusto.serialization.UserInfo, 111
 - es.deusto.server.data.GenericUser, 59
- getEventName
 - es.deusto.serialization.PostInfo, 80
 - es.deusto.server.data.Post, 75
- getEvents
 - es.deusto.server.dao.EventDAO, 49
- getEventsByOrganizer
 - es.deusto.server.dao.EventDAO, 50
- getEventsbyUser
 - es.deusto.server.dao.EventDAO, 50
- getInstance
 - es.deusto.server.dao.DAOFactory, 44
- getInterested
 - es.deusto.serialization.EventInfo, 53
 - es.deusto.server.data.Event, 46
- getInterests
 - es.deusto.serialization.SignupAttempt, 97
 - es.deusto.serialization.UserInfo, 112
 - es.deusto.server.data.User, 106
- getLanguage
 - es.deusto.client.windows.LanguageManager, 62
- getLanguageManager
 - es.deusto.client.controller.Controller, 35
- getName
 - es.deusto.serialization.EventInfo, 53
 - es.deusto.serialization.OrganizerInfo, 73
 - es.deusto.serialization.SignupAttempt, 98
 - es.deusto.serialization.TopicInfo, 104
 - es.deusto.serialization.UserInfo, 112
 - es.deusto.server.data.Event, 46
 - es.deusto.server.data.GenericUser, 59
 - es.deusto.server.data.Topic, 100
- getOrganization
 - es.deusto.serialization.OrganizerInfo, 73
 - es.deusto.serialization.SignupAttempt, 98
 - es.deusto.server.data.Organizer, 68
- getOrganize
 - es.deusto.client.controller.Controller, 35
- getOrganizer
 - es.deusto.server.dao.OrganizerDAO, 70
 - es.deusto.server.data.Event, 46
- getOrganizerEmail

- es.deusto.serialization.EventInfo, 53
 - es.deusto.serialization.PostInfo, 80
 - es.deusto.server.data.Post, 75
- getPassword
 - es.deusto.serialization.LoginAttempt, 64
 - es.deusto.serialization.SignupAttempt, 98
 - es.deusto.server.data.GenericUser, 59
- getPersistenceManager
 - es.deusto.server.dao.EventDAO, 50
 - es.deusto.server.dao.OrganizerDAO, 70
 - es.deusto.server.dao.PostDAO, 78
 - es.deusto.server.dao.TopicDAO, 102
 - es.deusto.server.dao.UserDAO, 108
- getPosts
 - es.deusto.serialization.EventInfo, 53
 - es.deusto.server.data.Event, 46
- getPostsByEvent
 - es.deusto.server.dao.PostDAO, 78
- getRecommendations
 - es.deusto.client.controller.Controller, 35
- getRecommendedEvents
 - es.deusto.server.server.AppService, 25
 - es.deusto.server.server.Server, 92
- getSavedEvents
 - es.deusto.serialization.SignupAttempt, 98
 - es.deusto.serialization.UserInfo, 112
 - es.deusto.server.data.User, 106
- getString
 - es.deusto.client.windows.LanguageManager, 62
- getTitle
 - es.deusto.serialization.PostInfo, 80
 - es.deusto.server.data.Post, 76
- getTopic
 - es.deusto.serialization.EventInfo, 54
 - es.deusto.server.dao.TopicDAO, 102
 - es.deusto.server.data.Event, 46
- getUser
 - es.deusto.client.controller.Controller, 35
 - es.deusto.server.dao.UserDAO, 108
- hello
 - es.deusto.server.server.AppService, 25
- id
 - es.deusto.client.windows.LogInWindow, 66
- isOrganizer
 - es.deusto.serialization.LoginAttempt, 64
- LanguageManager
 - es.deusto.client.windows.LanguageManager, 61
- LoginAttempt
 - es.deusto.serialization.LoginAttempt, 63
- LogInWindow
 - es.deusto.client.windows.LogInWindow, 65
- main
 - es.deusto.client.main.Main, 66
- Organizer
 - es.deusto.server.data.Organizer, 67
- OrganizerDAO
 - es.deusto.server.dao.OrganizerDAO, 69
- OrganizerEvents
 - es.deusto.client.windows.OrganizerEvents, 71
- OrganizerHome
 - es.deusto.client.windows.OrganizerHome, 72
- OrganizerInfo
 - es.deusto.serialization.OrganizerInfo, 73
- parseTopic
 - es.deusto.serialization.TopicInfo, 104
- Post
 - es.deusto.server.data.Post, 75
- PostCreationWindow
 - es.deusto.client.windows.PostCreationWindow, 77
- PostDAO
 - es.deusto.server.dao.PostDAO, 78
- PostInfo
 - es.deusto.serialization.PostInfo, 80
- Profile
 - es.deusto.client.windows.Profile, 82
- recoverPassword
 - es.deusto.server.server.AppService, 25
 - es.deusto.server.server.Server, 92
- reduceInterested
 - es.deusto.serialization.EventInfo, 54
 - es.deusto.server.data.Event, 46
- rule
 - es.deusto.junit.client.ControllerTest, 38
- SavedEventsWindow
 - es.deusto.client.windows.SavedEventsWindow, 83
- sayHello
 - es.deusto.server.server.Server, 93
- Server
 - es.deusto.server.server.Server, 87
- setCity
 - es.deusto.serialization.EventInfo, 54
 - es.deusto.serialization.SignupAttempt, 98
 - es.deusto.serialization.UserInfo, 112
 - es.deusto.server.data.Event, 46
 - es.deusto.server.data.User, 106
- setCountry
 - es.deusto.serialization.EventInfo, 54
 - es.deusto.serialization.SignupAttempt, 98
 - es.deusto.serialization.UserInfo, 112
 - es.deusto.server.data.Event, 46
 - es.deusto.server.data.User, 106
- setCreatedEvents
 - es.deusto.serialization.OrganizerInfo, 73
 - es.deusto.server.data.Organizer, 68
- setDate
 - es.deusto.serialization.EventInfo, 54
 - es.deusto.serialization.PostInfo, 81
 - es.deusto.server.data.Event, 47
 - es.deusto.server.data.Post, 76
- setDescription

- es.deusto.serialization.EventInfo, 54
 - es.deusto.serialization.PostInfo, 81
 - es.deusto.server.data.Event, 47
 - es.deusto.server.data.Post, 76
- setEmail
 - es.deusto.serialization.LoginAttempt, 64
 - es.deusto.serialization.OrganizerInfo, 74
 - es.deusto.serialization.SignupAttempt, 98
 - es.deusto.serialization.UserInfo, 112
 - es.deusto.server.data.GenericUser, 59
- setEventName
 - es.deusto.serialization.PostInfo, 81
 - es.deusto.server.data.Post, 76
- setInterested
 - es.deusto.serialization.EventInfo, 54
 - es.deusto.server.data.Event, 47
- setInterests
 - es.deusto.serialization.SignupAttempt, 99
 - es.deusto.serialization.UserInfo, 112
 - es.deusto.server.data.User, 107
- setLanguage
 - es.deusto.client.windows.LanguageManager, 62
- setName
 - es.deusto.serialization.EventInfo, 55
 - es.deusto.serialization.OrganizerInfo, 74
 - es.deusto.serialization.SignupAttempt, 99
 - es.deusto.serialization.TopicInfo, 104
 - es.deusto.serialization.UserInfo, 113
 - es.deusto.server.data.Event, 47
 - es.deusto.server.data.GenericUser, 59
 - es.deusto.server.data.Topic, 100
- setOrganization
 - es.deusto.serialization.OrganizerInfo, 74
 - es.deusto.serialization.SignupAttempt, 99
 - es.deusto.server.data.Organizer, 68
- setOrganizer
 - es.deusto.client.controller.Controller, 36
 - es.deusto.serialization.LoginAttempt, 64
 - es.deusto.server.data.Event, 47
- setOrganizerEmail
 - es.deusto.serialization.EventInfo, 55
 - es.deusto.serialization.PostInfo, 81
 - es.deusto.server.data.Post, 76
- setPassword
 - es.deusto.serialization.LoginAttempt, 64
 - es.deusto.serialization.SignupAttempt, 99
 - es.deusto.server.data.GenericUser, 60
- setPosts
 - es.deusto.serialization.EventInfo, 55
 - es.deusto.server.data.Event, 47
- setSavedEvents
 - es.deusto.serialization.UserInfo, 113
 - es.deusto.server.data.User, 107
- setSaveEvents
 - es.deusto.serialization.SignupAttempt, 99
- setTitle
 - es.deusto.serialization.PostInfo, 81
 - es.deusto.server.data.Post, 76
- setTopic
 - es.deusto.serialization.EventInfo, 55
 - es.deusto.server.data.Event, 47
- setUp
 - es.deusto.junit.server.ServerTest, 94
- setUpChild
 - es.deusto.junit.client.ControllerTest, 37
- setUser
 - es.deusto.client.controller.Controller, 36
- SignupAttempt
 - es.deusto.serialization.SignupAttempt, 96
- src/main/java/es/deusto/client/controller/Controller.java, 115
- src/main/java/es/deusto/client/main/Main.java, 115
- src/main/java/es/deusto/client/windows/ChangePass.java, 115
- src/main/java/es/deusto/client/windows/CodeVerification.java, 116
- src/main/java/es/deusto/client/windows/CreateEvent.java, 116
- src/main/java/es/deusto/client/windows/CreateOrganizerWindow.java, 116
- src/main/java/es/deusto/client/windows/CreateUserWindow.java, 117
- src/main/java/es/deusto/client/windows/EventListItem.java, 117
- src/main/java/es/deusto/client/windows/EventOrganizerWindow.java, 117
- src/main/java/es/deusto/client/windows/EventWindow.java, 117
- src/main/java/es/deusto/client/windows/ForgottenPassword.java, 118
- src/main/java/es/deusto/client/windows/LanguageManager.java, 118
- src/main/java/es/deusto/client/windows/LogInWindow.java, 118
- src/main/java/es/deusto/client/windows/OrganizerEvents.java, 119
- src/main/java/es/deusto/client/windows/OrganizerHome.java, 119
- src/main/java/es/deusto/client/windows/PostCreationWindow.java, 119
- src/main/java/es/deusto/client/windows/Profile.java, 119
- src/main/java/es/deusto/client/windows/SavedEventsWindow.java, 120
- src/main/java/es/deusto/client/windows/UserEventsWindow.java, 120
- src/main/java/es/deusto/serialization/EventInfo.java, 120
- src/main/java/es/deusto/serialization/GenericUserInfo.java, 121
- src/main/java/es/deusto/serialization/LoginAttempt.java, 121
- src/main/java/es/deusto/serialization/OrganizerInfo.java, 121
- src/main/java/es/deusto/serialization/PostInfo.java, 121
- src/main/java/es/deusto/serialization/SignupAttempt.java, 122
- src/main/java/es/deusto/serialization/TopicInfo.java, 122

- src/main/java/es/deusto/serialization/UserInfo.java, 122
- src/main/java/es/deusto/server/dao/DAOFactory.java, 122
- src/main/java/es/deusto/server/dao/EventDAO.java, 123
- src/main/java/es/deusto/server/dao/OrganizerDAO.java, 123
- src/main/java/es/deusto/server/dao/PostDAO.java, 123
- src/main/java/es/deusto/server/dao/TopicDAO.java, 123
- src/main/java/es/deusto/server/dao/UserDAO.java, 124
- src/main/java/es/deusto/server/data/Event.java, 124
- src/main/java/es/deusto/server/data/GenericUser.java, 124
- src/main/java/es/deusto/server/data/Organizer.java, 124
- src/main/java/es/deusto/server/data/Post.java, 125
- src/main/java/es/deusto/server/data/Topic.java, 125
- src/main/java/es/deusto/server/data/User.java, 125
- src/main/java/es/deusto/server/server/AppService.java, 125
- src/main/java/es/deusto/server/server/Server.java, 126
- src/test/java/es/deusto/junit/client/ControllerTest.java, 126
- src/test/java/es/deusto/junit/serialization/SerializationTest.java, 126
- src/test/java/es/deusto/junit/server/ServerTest.java, 126
- startUp
 - es.deusto.junit.client.ControllerTest, 37
 - es.deusto.junit.serialization.SerializationTest, 84
 - es.deusto.junit.server.ServerTest, 94
- storeEvent
 - es.deusto.server.dao.EventDAO, 51
- storeOrganizer
 - es.deusto.server.dao.OrganizerDAO, 70
- storePost
 - es.deusto.server.dao.PostDAO, 79
- storeTopic
 - es.deusto.server.dao.TopicDAO, 102
- storeUser
 - es.deusto.server.dao.UserDAO, 108
- suite
 - es.deusto.junit.client.ControllerTest, 37
 - es.deusto.junit.serialization.SerializationTest, 84
 - es.deusto.junit.server.ServerTest, 94
- tearDown
 - es.deusto.junit.server.ServerTest, 94
- tearDownChild
 - es.deusto.junit.client.ControllerTest, 37
- testCreateEvent
 - es.deusto.junit.server.ServerTest, 94
- testCreatePost
 - es.deusto.junit.server.ServerTest, 94
- testEventInfoFromEvent
 - es.deusto.junit.serialization.SerializationTest, 84
- testEventInfoSetters
 - es.deusto.junit.serialization.SerializationTest, 84
- testNormalLogin
 - es.deusto.junit.client.ControllerTest, 38
- testNormalSignup
 - es.deusto.junit.server.ServerTest, 95
- testNormalSignupAndDelete
 - es.deusto.junit.client.ControllerTest, 38
- testNormalUpdate
 - es.deusto.junit.server.ServerTest, 95
- testOrganizerInfoFromOrganizer
 - es.deusto.junit.serialization.SerializationTest, 84
- testOrganizerInfoSetters
 - es.deusto.junit.serialization.SerializationTest, 84
- testOrganizerLogin
 - es.deusto.junit.client.ControllerTest, 38
 - es.deusto.junit.server.ServerTest, 95
- testOrganizerSignup
 - es.deusto.junit.server.ServerTest, 95
- testOrganizerSignupAndDelete
 - es.deusto.junit.client.ControllerTest, 38
- testOrganizerUpdate
 - es.deusto.junit.server.ServerTest, 95
- testPostInfoFromPost
 - es.deusto.junit.serialization.SerializationTest, 85
- testPostInfoSetters
 - es.deusto.junit.serialization.SerializationTest, 85
- testSignupAttemptOrganizer
 - es.deusto.junit.serialization.SerializationTest, 85
- testSignupAttemptUser
 - es.deusto.junit.serialization.SerializationTest, 85
- testUserInfoFromUser
 - es.deusto.junit.serialization.SerializationTest, 85
- testUserInfoSetters
 - es.deusto.junit.serialization.SerializationTest, 85
- testUserInfoStringConstructor
 - es.deusto.junit.serialization.SerializationTest, 85
- testUserUpdate
 - es.deusto.junit.client.ControllerTest, 38
- Topic
 - es.deusto.server.data.Topic, 100
- TopicDAO
 - es.deusto.server.dao.TopicDAO, 101
- TopicInfo
 - es.deusto.serialization.TopicInfo, 103, 104
- toString
 - es.deusto.serialization.LoginAttempt, 64
 - es.deusto.serialization.SignupAttempt, 99
 - es.deusto.serialization.TopicInfo, 104
 - es.deusto.server.data.Event, 48
 - es.deusto.server.data.GenericUser, 60
 - es.deusto.server.data.Organizer, 68
 - es.deusto.server.data.Topic, 101
 - es.deusto.server.data.User, 107
- updateEvent
 - es.deusto.server.dao.EventDAO, 51
- updateOrganizer
 - es.deusto.server.dao.OrganizerDAO, 70
- updatePost
 - es.deusto.server.dao.PostDAO, 79
- updateTopic
 - es.deusto.server.dao.TopicDAO, 102
- updateUser
 - es.deusto.server.dao.UserDAO, 109

- [es.deusto.server.server.AppService](#), [26](#)
- UserDAO
 - [es.deusto.server.dao.UserDAO](#), [108](#)
- UserEventsWindow
 - [es.deusto.client.windows.UserEventsWindow](#), [109](#)
- UserInfo
 - [es.deusto.serialization.UserInfo](#), [111](#)