FactDev

0.1

Generated by Doxygen 1.8.9.1

Mon Mar 30 2015 22:03:59

Contents

1 FactDev documentation 1

2 README 3

3 QTestRunner 5

4 Namespace Index 7

4.1 Namespace List . . . 7

5 Hierarchical Index 9

5.1 Class Hierarchy . . . 9

6 Class Index 13

6.1 Class List . . . 13

7 Namespace Documentation 17

7.1 Databases Namespace Reference . . . 17

7.1.1 Detailed Description . . . 17

7.2 Gui Namespace Reference . . . 17

7.2.1 Detailed Description . . . 18

7.3 Gui::Widgets Namespace Reference . . . 18

7.3.1 Detailed Description . . . 18

7.4 Models Namespace Reference . . . 18

7.4.1 Detailed Description . . . 19

8 Class Documentation 21

8.1 Databases::AccessDatabase Class Reference . . . 21

8.2 Gui::Dialogs::AddProjectDialog Class Reference . . . 21

8.2.1 Detailed Description . . . 22

8.2.2 Constructor & Destructor Documentation . . . 22

8.2.2.1 AddProjectDialog . . . 22

8.3 Gui::Dialogs::AddQuoteDialog Class Reference . . . 22

8.3.1 Detailed Description . . . 23

8.3.2 Constructor & Destructor Documentation . . . 23

8.3.2.1 AddQuoteDialog . . . 23
8.3.3 Member Function Documentation . . . 24
8.3.3.1 fillQuoteBilling . . . 24
8.3.3.2 getCopy . . . 24
8.3.3.3 getIdCustomer . . . 24
8.3.3.4 getNumber . . . 24
8.3.3.5 setCopy . . . 24
8.3.3.6 setQuoteIdNumber . . . 25
8.4 Models::Billing Class Reference . . . 25
8.4.1 Detailed Description . . . 27
8.4.2 Constructor & Destructor Documentation . . . 27
8.4.2.1 Billing . . . 27
8.4.3 Member Function Documentation . . . 27
8.4.3.1 addContributory . . . 27
8.4.3.2 getContributories . . . 27
8.4.3.3 getDataMap . . . 27
8.4.3.4 getDate . . . 27
8.4.3.5 getDescription . . . 28
8.4.3.6 getFilename . . . 28
8.4.3.7 getFolder . . . 28
8.4.3.8 getItem . . . 28
8.4.3.9 getNumber . . . 28
8.4.3.10 getPath . . . 28
8.4.3.11 getPrice . . . 29
8.4.3.12 getSumQuantity . . . 29
8.4.3.13 getTitle . . . 29
8.4.3.14 hydrat . . . 29
8.4.3.15 isBilling . . . 29
8.4.3.16 isPaid . . . 29
8.4.3.17 operator"!= . . . 29
8.4.3.18 operator< . . . 30
8.4.3.19 operator== . . . 30
8.4.3.20 setDate . . . 30
8.4.3.21 setDescription . . . 30
8.4.3.22 setIsBilling . . . 30
8.4.3.23 setIsPaid . . . 31
8.4.3.24 setNumber . . . 31
8.4.3.25 setTitle . . . 31
8.5 Databases::BillingDatabase Class Reference . . . 31
8.5.1 Detailed Description . . . 32

8.5.2 Member Function Documentation . . . 33
8.5.2.1 addBilling . . . 33
8.5.2.2 addBillingProject . . . 33
8.5.2.3 getAllBillingsOfProject . . . 33
8.5.2.4 getAllBillingsOnly . . . 33
8.5.2.5 getBilling . . . 33
8.5.2.6 getBilling . . . 33
8.5.2.7 getBillings . . . 34
8.5.2.8 getBillingsBetweenDates . . . 34
8.5.2.9 getBillingsTable . . . 34
8.5.2.10 getMaxBillingNumber . . . 34
8.5.2.11 getMaxBillingNumberOfCustomer . . . 35
8.5.2.12 getMaxQuoteNumber . . . 36
8.5.2.13 getMaxQuoteNumberOfCustomer . . . 36
8.5.2.14 instance . . . 36
8.5.2.15 isBillingPaid . . . 36
8.5.2.16 removeBilling . . . 36
8.5.2.17 removeBillingProject . . . 37
8.6 BillingDatabaseTest Class Reference . . . 37
8.7 BillingModelTest Class Reference . . . 37
8.8 Gui::Widgets::WdgModels::BillingsTableModel Class Reference . . . 38
8.8.1 Detailed Description . . . 38
8.8.2 Constructor & Destructor Documentation . . . 39
8.8.2.1 BillingsTableModel . . . 39
8.8.3 Member Function Documentation . . . 40
8.8.3.1 append . . . 40
8.8.3.2 columnCount . . . 40
8.8.3.3 count . . . 40
8.8.3.4 data . . . 40
8.8.3.5 flags . . . 40
8.8.3.6 getBillings . . . 41
8.8.3.7 headerData . . . 41
8.8.3.8 remove . . . 41
8.8.3.9 rowCount . . . 41
8.8.3.10 setData . . . 41
8.9 Models::Calculable Class Reference . . . 42
8.9.1 Detailed Description . . . 42
8.9.2 Member Function Documentation . . . 42
8.9.2.1 getPrice . . . 42
8.9.2.2 getSumQuantity . . . 42

8.10 Gui::Widgets::CheckFields::CheckCity Class Reference . . . 43
8.10.1 Detailed Description . . . 43
8.10.2 Constructor & Destructor Documentation . . . 43
8.10.2.1 CheckCity . . . 43
8.11 Gui::Widgets::CheckFields::CheckCountry Class Reference . . . 43
8.11.1 Detailed Description . . . 44
8.11.2 Constructor & Destructor Documentation . . . 44
8.11.2.1 CheckCountry . . . 44
8.12 Gui::Widgets::CheckFields::CheckEmail Class Reference . . . 44
8.12.1 Detailed Description . . . 45
8.12.2 Constructor & Destructor Documentation . . . 45
8.12.2.1 CheckEmail . . . 45
8.12.3 Member Function Documentation . . . 45
8.12.3.1 check . . . 45
8.13 Gui::Widgets::CheckFields::CheckFieldsLetters Class Reference . . . 46
8.13.1 Detailed Description . . . 46
8.13.2 Constructor & Destructor Documentation . . . 46
8.13.2.1 CheckFieldsLetters . . . 46
8.13.3 Member Function Documentation . . . 46
8.13.3.1 check . . . 46
8.14 Gui::Widgets::CheckFields::CheckFieldsNumbers Class Reference . . . 47
8.14.1 Detailed Description . . . 47
8.14.2 Constructor & Destructor Documentation . . . 47
8.14.2.1 CheckFieldsNumbers . . . 47
8.14.3 Member Function Documentation . . . 48
8.14.3.1 check . . . 48
8.15 Gui::Widgets::CheckFields::CheckIpAddress Class Reference . . . 48
8.15.1 Detailed Description . . . 48
8.15.2 Constructor & Destructor Documentation . . . 49
8.15.2.1 CheckIpAddress . . . 49
8.15.3 Member Function Documentation . . . 49
8.15.3.1 check . . . 49
8.16 Gui::Widgets::CheckFields::CheckLogin Class Reference . . . 49
8.16.1 Detailed Description . . . 50
8.16.2 Constructor & Destructor Documentation . . . 50
8.16.2.1 CheckLogin . . . 50
8.16.3 Member Function Documentation . . . 50
8.16.3.1 check . . . 50
8.17 Gui::Widgets::CheckFields::CheckName Class Reference . . . 51
8.17.1 Detailed Description . . . 51

8.17.2 Constructor & Destructor Documentation . . . 51
8.17.2.1 CheckName . . . 51
8.18 Gui::Widgets::CheckFields::CheckPhone Class Reference . . . 51
8.18.1 Detailed Description . . . 52
8.18.2 Constructor & Destructor Documentation . . . 52
8.18.2.1 CheckPhone . . . 52
8.18.3 Member Function Documentation . . . 52
8.18.3.1 check . . . 52
8.18.3.2 getCountry . . . 53
8.18.3.3 setCountry . . . 53
8.19 Gui::Widgets::CheckFields::CheckPortNumber Class Reference . . . 53
8.19.1 Detailed Description . . . 53
8.19.2 Constructor & Destructor Documentation . . . 54
8.19.2.1 CheckPortNumber . . . 54
8.19.3 Member Function Documentation . . . 54
8.19.3.1 check . . . 54
8.20 Gui::Widgets::CheckFields::CheckPostalCode Class Reference . . . 54
8.20.1 Detailed Description . . . 55
8.20.2 Constructor & Destructor Documentation . . . 55
8.20.2.1 CheckPostalCode . . . 55
8.20.3 Member Function Documentation . . . 55
8.20.3.1 check . . . 55
8.20.3.2 getCountry . . . 55
8.20.3.3 setCountry . . . 55
8.21 Gui::Widgets::CheckFields::CheckQLineEdit Class Reference . . . 56
8.21.1 Detailed Description . . . 56
8.21.2 Constructor & Destructor Documentation . . . 56
8.21.2.1 CheckQLineEdit . . . 56
8.21.3 Member Function Documentation . . . 57
8.21.3.1 getBtnValid . . . 57
8.21.3.2 isValid . . . 57
8.21.3.3 setBtnValid . . . 57
8.22 Gui::Widgets::CheckFields::CheckSiretNumber Class Reference . . . 57
8.22.1 Detailed Description . . . 58
8.22.2 Constructor & Destructor Documentation . . . 58
8.22.2.1 CheckSiretNumber . . . 58
8.22.3 Member Function Documentation . . . 58
8.22.3.1 check . . . 58
8.23 Gui::Widgets::CheckFields::CheckUntilField Class Reference . . . 58
8.23.1 Detailed Description . . . 59

8.23.2 Constructor & Destructor Documentation . . . 59
8.23.2.1 CheckUntilField . . . 59
8.23.3 Member Function Documentation . . . 59
8.23.3.1 check . . . 59
8.24 Gui::Widgets::CheckFields::CheckValidField Class Reference . . . 59
8.24.1 Detailed Description . . . 60
8.24.2 Constructor & Destructor Documentation . . . 60
8.24.2.1 CheckValidField . . . 60
8.24.3 Member Function Documentation . . . 60
8.24.3.1 check . . . 60
8.25 Gui::Widgets::CheckFields::CheckWebsite Class Reference . . . 61
8.25.1 Detailed Description . . . 61
8.25.2 Constructor & Destructor Documentation . . . 61
8.25.2.1 CheckWebsite . . . 61
8.25.3 Member Function Documentation . . . 61
8.25.3.1 check . . . 61
8.26 Gui::Widgets::Path::ChoseDirectoryWidget Class Reference . . . 62
8.26.1 Detailed Description . . . 62
8.26.2 Constructor & Destructor Documentation . . . 62
8.26.2.1 ChoseDirectoryWidget . . . 62
8.27 Gui::Widgets::Path::ChoseFileWidget Class Reference . . . 63
8.27.1 Detailed Description . . . 63
8.27.2 Constructor & Destructor Documentation . . . 63
8.27.2.1 ChoseFileWidget . . . 63
8.27.3 Member Function Documentation . . . 64
8.27.3.1 getypeFiles . . . 64
8.27.3.2 setTypeFiles . . . 64
8.28 Gui::Widgets::Path::ChosePathWidget Class Reference . . . 64
8.28.1 Detailed Description . . . 65
8.28.2 Constructor & Destructor Documentation . . . 65
8.28.2.1 ChosePathWidget . . . 65
8.28.3 Member Function Documentation . . . 65
8.28.3.1 getDefaultLocation . . . 65
8.28.3.2 getField . . . 65
8.28.3.3 setField . . . 65
8.29 Gui::Widgets::Delegates::ComboBoxDelegate Class Reference . . . 66
8.29.1 Detailed Description . . . 66
8.29.2 Constructor & Destructor Documentation . . . 67
8.29.2.1 ComboBoxDelegate . . . 67
8.29.3 Member Function Documentation . . . 67

8.29.3.1 createEditor . . . 67
8.29.3.2 paint . . . 67
8.29.3.3 setEditorData . . . 67
8.29.3.4 setModelData . . . 68
8.29.3.5 updateEditorGeometry . . . 69
8.30 Gui::Widgets::ComboBoxModelWidget Class Reference . . . 69
8.30.1 Detailed Description . . . 69
8.30.2 Constructor & Destructor Documentation . . . 69
8.30.2.1 ComboBoxModelWidget . . . 69
8.31 Gui::Dialogs::ComputeTurnoverDialog Class Reference . . . 70
8.31.1 Detailed Description . . . 70
8.31.2 Member Function Documentation . . . 70
8.31.2.1 beginDateControl . . . 70
8.31.2.2 endDateControl . . . 71
8.31.2.3 fillLabels . . . 71
8.32 Mustache::Context Class Reference . . . 71
8.32.1 Detailed Description . . . 72
8.32.2 Constructor & Destructor Documentation . . . 72
8.32.2.1 Context . . . 72
8.32.3 Member Function Documentation . . . 72
8.32.3.1 canEval . . . 72
8.32.3.2 eval . . . 72
8.32.3.3 isFalse . . . 72
8.32.3.4 listCount . . . 72
8.32.3.5 partialResolver . . . 72
8.32.3.6 partialValue . . . 72
8.32.3.7 pop . . . 73
8.32.3.8 push . . . 73
8.32.3.9 stringValue . . . 73
8.33 ContributoriesDatabaseTest Class Reference . . . 73
8.34 Models::ContributoriesList Class Reference . . . 73
8.34.1 Detailed Description . . . 74
8.34.2 Member Function Documentation . . . 75
8.34.2.1 addContributory . . . 75
8.34.2.2 addProject . . . 76
8.34.2.3 getAllContributories . . . 76
8.34.2.4 getContributories . . . 76
8.34.2.5 getCustomer . . . 76
8.34.2.6 getDataMap . . . 76
8.34.2.7 getIdBilling . . . 76

8.34.2.8 getNbProjects . . . 77
8.34.2.9 getPrice . . . 77
8.34.2.10 getPrice . . . 77
8.34.2.11 getProjects . . . 77
8.34.2.12 getRate . . . 77
8.34.2.13 getSumQuantity . . . 77
8.34.2.14 getSumQuantity . . . 78
8.34.2.15 isInsert . . . 78
8.34.2.16 setAllIdContributories . . . 78
8.34.2.17 setIdBilling . . . 78
8.34.2.18 setInsert . . . 78
8.35 Gui::Widgets::WdgModels::ContributoriesTableModel Class Reference . . . 78
8.35.1 Detailed Description . . . 79
8.35.2 Constructor & Destructor Documentation . . . 79
8.35.2.1 ContributoriesTableModel . . . 79
8.35.3 Member Function Documentation . . . 80
8.35.3.1 append . . . 80
8.35.3.2 columnCount . . . 80
8.35.3.3 count . . . 80
8.35.3.4 data . . . 80
8.35.3.5 flags . . . 80
8.35.3.6 getContributories . . . 81
8.35.3.7 getSumQuantity . . . 81
8.35.3.8 headerData . . . 81
8.35.3.9 remove . . . 81
8.35.3.10 rowCount . . . 81
8.35.3.11 setData . . . 81
8.36 Gui::Widgets::ContributoriesWidget Class Reference . . . 82
8.36.1 Detailed Description . . . 83
8.36.2 Constructor & Destructor Documentation . . . 83
8.36.2.1 ContributoriesWidget . . . 83
8.36.3 Member Function Documentation . . . 83
8.36.3.1 add . . . 83
8.36.3.2 addProject . . . 83
8.36.3.3 count . . . 83
8.36.3.4 getContributories . . . 84
8.37 Models::Contributory Class Reference . . . 84
8.37.1 Detailed Description . . . 85
8.37.2 Constructor & Destructor Documentation . . . 85
8.37.2.1 Contributory . . . 85

8.37.3 Member Function Documentation . . . 85
8.37.3.1 getDataMap . . . 85
8.37.3.2 getDescription . . . 86
8.37.3.3 getHourlyRate . . . 86
8.37.3.4 getLongDescription . . . 86
8.37.3.5 getPrice . . . 86
8.37.3.6 getProject . . . 86
8.37.3.7 getQuantity . . . 86
8.37.3.8 getSumQuantity . . . 87
8.37.3.9 getUnit . . . 87
8.37.3.10 hydrat . . . 87
8.37.3.11 operator"!= . . . 87
8.37.3.12 operator== . . . 87
8.37.3.13 setDescription . . . 87
8.37.3.14 setHourlyRate . . . 88
8.37.3.15 setLongDescription . . . 88
8.37.3.16 setProject . . . 88
8.37.3.17 setQuantity . . . 88
8.37.3.18 setUnit . . . 88
8.38 Databases::ContributoryDatabase Class Reference . . . 88
8.38.1 Detailed Description . . . 89
8.38.2 Member Function Documentation . . . 89
8.38.2.1 addContributory . . . 89
8.38.2.2 getContributoriesByBilling . . . 90
8.38.2.3 getContributoriesByBillingAndProject . . . 90
8.38.2.4 getContributory . . . 90
8.38.2.5 getContributory . . . 90
8.38.2.6 instance . . . 90
8.38.2.7 removeContributory . . . 91
8.39 ContributoryListTest Class Reference . . . 91
8.40 ContributoryModelTest Class Reference . . . 91
8.41 CounterContext Class Reference . . . 92
8.41.1 Member Function Documentation . . . 92
8.41.1.1 canEval . . . 92
8.41.1.2 canEval . . . 92
8.41.1.3 eval . . . 93
8.41.1.4 eval . . . 93
8.41.1.5 stringValue . . . 93
8.41.1.6 stringValue . . . 93
8.42 Models::Customer Class Reference . . . 93

8.42.1 Detailed Description . . . 94
8.42.2 Constructor & Destructor Documentation . . . 94
8.42.2.1 Customer . . . 94
8.42.3 Member Function Documentation . . . 94
8.42.3.1 getDataMap . . . 94
8.42.3.2 getNameFolder . . . 94
8.42.3.3 getPath . . . 95
8.42.3.4 getTurnover . . . 95
8.42.3.5 hydrat . . . 95
8.43 Gui::Widgets::CustomerContextualMenu Class Reference . . . 95
8.43.1 Detailed Description . . . 96
8.43.2 Constructor & Destructor Documentation . . . 96
8.43.2.1 CustomerContextualMenu . . . 96
8.44 Databases::CustomerDatabase Class Reference . . . 96
8.44.1 Detailed Description . . . 97
8.44.2 Member Function Documentation . . . 97
8.44.2.1 addCustomer . . . 97
8.44.2.2 getCustomer . . . 97
8.44.2.3 getCustomer . . . 98
8.44.2.4 getCustomersTable . . . 99
8.44.2.5 getItemCustomer . . . 99
8.44.2.6 getItemProject . . . 99
8.44.2.7 getItemRoot . . . 99
8.44.2.8 getNbCustomers . . . 100
8.44.2.9 getTree . . . 100
8.44.2.10 instance . . . 100
8.44.2.11 removeCustomer . . . 100
8.44.2.12 updateCustomer . . . 100
8.45 CustomerDatabaseTest Class Reference . . . 101
8.46 Gui::Widgets::CustomerDataWidget Class Reference . . . 101
8.46.1 Detailed Description . . . 101
8.46.2 Constructor & Destructor Documentation . . . 101
8.46.2.1 CustomerDataWidget . . . 101
8.46.3 Member Function Documentation . . . 102
8.46.3.1 printInformations . . . 102
8.47 CustomerModelTest Class Reference . . . 102
8.48 Gui::Widgets::WdgModels::CustomersTableModel Class Reference . . . 102
8.48.1 Detailed Description . . . 103
8.48.2 Constructor & Destructor Documentation . . . 103
8.48.2.1 CustomersTableModel . . . 103

8.48.3 Member Function Documentation . . . 103
8.48.3.1 append . . . 103
8.48.3.2 columnCount . . . 104
8.48.3.3 count . . . 104
8.48.3.4 data . . . 104
8.48.3.5 flags . . . 104
8.48.3.6 getCustomers . . . 104
8.48.3.7 headerData . . . 105
8.48.3.8 remove . . . 106
8.48.3.9 rowCount . . . 106
8.48.3.10 setData . . . 106
8.49 Databases::Database Class Reference . . . 106
8.49.1 Detailed Description . . . 108
8.49.2 Member Function Documentation . . . 108
8.49.2.1 changeDatabase . . . 108
8.49.2.2 executeFile . . . 108
8.49.2.3 instance . . . 108
8.49.2.4 lastError . . . 108
8.49.2.5 setDatabase . . . 109
8.49.2.6 value . . . 109
8.50 Gui::Widgets::DatabaseSettingsWidget Class Reference . . . 109
8.50.1 Detailed Description . . . 110
8.50.2 Constructor & Destructor Documentation . . . 110
8.50.2.1 DatabaseSettingsWidget . . . 110
8.50.3 Member Function Documentation . . . 110
8.50.3.1 getDatabaseName . . . 110
8.50.3.2 getDomainNameOrIP . . . 111
8.50.3.3 getLogin . . . 111
8.50.3.4 getPassword . . . 111
8.50.3.5 getPort . . . 111
8.50.3.6 isValid . . . 111
8.51 Exceptions::DbException Class Reference . . . 111
8.51.1 Detailed Description . . . 112
8.51.2 Constructor & Destructor Documentation . . . 112
8.51.2.1 DbException . . . 112
8.51.3 Member Function Documentation . . . 112
8.51.3.1 popupMessage . . . 112
8.52 Gui::Dialogs::DialogAddCustomer Class Reference . . . 113
8.52.1 Detailed Description . . . 113
8.52.2 Constructor & Destructor Documentation . . . 113

8.52.2.1 DialogAddCustomer . . . 113
8.53 Utils::Directories Class Reference . . . 114
8.53.1 Member Function Documentation . . . 114
8.53.1.1 makeDirectory . . . 114
8.54 Utils::Double Class Reference . . . 114
8.54.1 Detailed Description . . . 114
8.54.2 Member Function Documentation . . . 114
8.54.2.1 round . . . 114
8.55 Gui::Widgets::Delegates::DoubleSpinBoxDelegate Class Reference . . . 115
8.55.1 Detailed Description . . . 115
8.55.2 Constructor & Destructor Documentation . . . 116
8.55.2.1 DoubleSpinBoxDelegate . . . 116
8.55.3 Member Function Documentation . . . 116
8.55.3.1 createEditor . . . 116
8.55.3.2 setEditorData . . . 116
8.55.3.3 setModelData . . . 116
8.55.3.4 updateEditorGeometry . . . 116
8.56 FileChoseWidget Class Reference . . . 117
8.57 Exceptions::FileException Class Reference . . . 117
8.57.1 Detailed Description . . . 117
8.57.2 Constructor & Destructor Documentation . . . 118
8.57.2.1 FileException . . . 118
8.57.3 Member Function Documentation . . . 119
8.57.3.1 popupMessage . . . 119
8.58 Generation Class Reference . . . 119
8.59 Utils::HierarchicalSystem Class Reference . . . 119
8.59.1 Detailed Description . . . 120
8.59.2 Member Function Documentation . . . 120
8.59.2.1 addBillingToProject . . . 120
8.59.2.2 addProjectToCustomer . . . 120
8.59.2.3 getCustomers . . . 120
8.59.2.4 getProjects . . . 121
8.60 Gui::Widgets::CheckFields::ICheckField Class Reference . . . 121
8.60.1 Detailed Description . . . 121
8.60.2 Member Function Documentation . . . 121
8.60.2.1 check . . . 121
8.61 Models::IModel Class Reference . . . 122
8.61.1 Detailed Description . . . 122
8.61.2 Member Function Documentation . . . 123
8.61.2.1 getDataMap . . . 123

8.61.2.2 getId . . . 123
8.61.2.3 hydrat . . . 123
8.61.2.4 isToRemoved . . . 123
8.61.2.5 setId . . . 123
8.61.2.6 setToRemoved . . . 123
8.62 Utils::ItemType Class Reference . . . 124
8.62.1 Detailed Description . . . 124
8.62.2 Constructor & Destructor Documentation . . . 124
8.62.2.1 ItemType . . . 124
8.62.3 Member Function Documentation . . . 125
8.62.3.1 getModel . . . 125
8.62.3.2 getName . . . 125
8.62.3.3 getType . . . 125
8.62.3.4 setName . . . 125
8.62.3.5 setType . . . 125
8.63 ItemTypeTest Class Reference . . . 126
8.64 Utils::Log Class Reference . . . 126
8.64.1 Detailed Description . . . 126
8.64.2 Member Function Documentation . . . 126
8.64.2.1 instance . . . 126
8.64.2.2 write . . . 127
8.64.3 Friends And Related Function Documentation . . . 127
8.64.3.1 operator<< . . . 127
8.65 Gui::MainWindow Class Reference . . . 127
8.65.1 Detailed Description . . . 129
8.65.2 Constructor & Destructor Documentation . . . 129
8.65.2.1 MainWindow . . . 129
8.65.3 Member Function Documentation . . . 129
8.65.3.1 addBill . . . 129
8.65.3.2 addDoc . . . 129
8.65.3.3 addProject . . . 130
8.65.3.4 addQuote . . . 130
8.65.3.5 editUser . . . 130
8.65.3.6 getCurrentCustomerId . . . 130
8.65.3.7 getCurrentCustomerName . . . 130
8.65.3.8 getCurrentProjectId . . . 130
8.65.3.9 getCurrentProjectName . . . 131
8.65.3.10 getCurrentQuoteId . . . 131
8.65.3.11 resizeEvent . . . 131
8.65.3.12 rootTree . . . 131

8.65.3.13 search . . . 131
8.65.3.14 treeLevel . . . 131
8.66 Gui::Dialogs::MessageBox Class Reference . . . 132
8.66.1 Detailed Description . . . 132
8.66.2 Constructor & Destructor Documentation . . . 132
8.66.2.1 MessageBox . . . 132
8.66.3 Member Function Documentation . . . 133
8.66.3.1 setImage . . . 133
8.66.3.2 setText . . . 133
8.67 Parameters Class Reference . . . 133
8.67.1 Detailed Description . . . 133
8.67.2 Member Data Documentation . . . 133
8.67.2.1 DB_FILENAME . . . 133
8.67.2.2 VERSION . . . 134
8.68 Mustache::PartialFileLoader Class Reference . . . 134
8.68.1 Detailed Description . . . 134
8.68.2 Member Function Documentation . . . 134
8.68.2.1 getPartial . . . 134
8.69 Mustache::PartialMap Class Reference . . . 134
8.69.1 Detailed Description . . . 135
8.69.2 Member Function Documentation . . . 135
8.69.2.1 getPartial . . . 135
8.70 Mustache::PartialResolver Class Reference . . . 135
8.70.1 Detailed Description . . . 135
8.70.2 Member Function Documentation . . . 136
8.70.2.1 getPartial . . . 136
8.71 Generator::PdfGenerator Class Reference . . . 136
8.71.1 Detailed Description . . . 136
8.71.2 Constructor & Destructor Documentation . . . 136
8.71.2.1 PdfGenerator . . . 136
8.71.3 Member Function Documentation . . . 136
8.71.3.1 generate . . . 136
8.72 Models::People Class Reference . . . 137
8.72.1 Detailed Description . . . 138
8.72.2 Member Function Documentation . . . 138
8.72.2.1 getAddress . . . 138
8.72.2.2 getAddressComplement . . . 138
8.72.2.3 getCity . . . 138
8.72.2.4 getCompany . . . 139
8.72.2.5 getCountry . . . 139

8.72.2.6 getEmail . . . 139
8.72.2.7 getFax . . . 139
8.72.2.8 getFirstname . . . 139
8.72.2.9 getLastname . . . 139
8.72.2.10 getMobilePhone . . . 139
8.72.2.11 getPhone . . . 140
8.72.2.12 getPostalCode . . . 140
8.72.2.13 getWebsite . . . 140
8.72.2.14 operator"!= . . . 140
8.72.2.15 operator== . . . 140
8.72.2.16 setAddress . . . 140
8.72.2.17 setAddressComplement . . . 141
8.72.2.18 setCity . . . 141
8.72.2.19 setCompany . . . 141
8.72.2.20 setCountry . . . 141
8.72.2.21 setEmail . . . 141
8.72.2.22 setFax . . . 141
8.72.2.23 setFirstname . . . 142
8.72.2.24 setLastname . . . 142
8.72.2.25 setMobilePhone . . . 142
8.72.2.26 setPhone . . . 142
8.72.2.27 setPostalCode . . . 142
8.72.2.28 setWebsite . . . 142
8.73 Utils::pointers Class Reference . . . 143
8.74 PointersTest Class Reference . . . 143
8.75 Gui::Widgets::Popup Class Reference . . . 143
8.75.1 Detailed Description . . . 143
8.76 Models::Project Class Reference . . . 144
8.76.1 Detailed Description . . . 145
8.76.2 Constructor & Destructor Documentation . . . 145
8.76.2.1 Project . . . 145
8.76.3 Member Function Documentation . . . 145
8.76.3.1 getBeginDate . . . 145
8.76.3.2 getCost . . . 146
8.76.3.3 getCost . . . 146
8.76.3.4 getCustomer . . . 146
8.76.3.5 getDailyRate . . . 146
8.76.3.6 getDataMap . . . 146
8.76.3.7 getDescription . . . 146
8.76.3.8 getEndDate . . . 147

8.76.3.9 getName . . . 147
8.76.3.10 hydrat . . . 147
8.76.3.11 operator"!= . . . 147
8.76.3.12 operator< . . . 147
8.76.3.13 operator== . . . 147
8.76.3.14 setBeginDate . . . 148
8.76.3.15 setCost . . . 148
8.76.3.16 setCustomer . . . 148
8.76.3.17 setDailyRate . . . 148
8.76.3.18 setDescription . . . 148
8.76.3.19 setEndDate . . . 148
8.76.3.20 setName . . . 149
8.77 Gui::Widgets::Delegates::ProjectComboDelegate Class Reference . . . 149
8.77.1 Detailed Description . . . 150
8.77.2 Constructor & Destructor Documentation . . . 150
8.77.2.1 ProjectComboDelegate . . . 150
8.77.3 Member Function Documentation . . . 150
8.77.3.1 createEditor . . . 150
8.77.3.2 getProjects . . . 150
8.77.3.3 isLocked . . . 150
8.77.3.4 paint . . . 150
8.77.3.5 setLocked . . . 151
8.78 Gui::Widgets::WdgModels::ProjectContributoriesTableModel Class Reference . . . 151
8.78.1 Detailed Description . . . 152
8.78.2 Constructor & Destructor Documentation . . . 152
8.78.2.1 ProjectContributoriesTableModel . . . 152
8.78.3 Member Function Documentation . . . 152
8.78.3.1 allProjectsChosen . . . 152
8.78.3.2 append . . . 152
8.78.3.3 columnCount . . . 153
8.78.3.4 data . . . 153
8.78.3.5 flags . . . 153
8.78.3.6 getProject . . . 153
8.78.3.7 getProjects . . . 153
8.78.3.8 getSelectedProjects . . . 154
8.78.3.9 headerData . . . 154
8.78.3.10 remove . . . 154
8.78.3.11 rowCount . . . 154
8.78.3.12 setData . . . 154
8.79 Databases::ProjectDatabase Class Reference . . . 155

8.79.1 Detailed Description . . . 156
8.79.2 Member Function Documentation . . . 156
8.79.2.1 addProject . . . 156
8.79.2.2 getAllProjects . . . 156
8.79.2.3 getCostProjects . . . 156
8.79.2.4 getNbProjects . . . 156
8.79.2.5 getNbProjectsForACustomer . . . 156
8.79.2.6 getProject . . . 157
8.79.2.7 getProject . . . 157
8.79.2.8 getProjects . . . 157
8.79.2.9 getProjectsOfCustomer . . . 157
8.79.2.10 getProjectsTable . . . 158
8.79.2.11 instance . . . 159
8.79.2.12 removeProject . . . 159
8.79.2.13 updateProject . . . 159
8.80 ProjectDatabaseTest Class Reference . . . 159
8.81 ProjectModelTest Class Reference . . . 160
8.82 Gui::Widgets::WdgModels::ProjectsTableModel Class Reference . . . 160
8.82.1 Detailed Description . . . 161
8.82.2 Constructor & Destructor Documentation . . . 161
8.82.2.1 ProjectsTableModel . . . 161
8.82.3 Member Function Documentation . . . 161
8.82.3.1 append . . . 161
8.82.3.2 columnCount . . . 161
8.82.3.3 count . . . 161
8.82.3.4 data . . . 161
8.82.3.5 flags . . . 162
8.82.3.6 getProjects . . . 162
8.82.3.7 headerData . . . 162
8.82.3.8 remove . . . 162
8.82.3.9 rowCount . . . 163
8.82.3.10 setData . . . 163
8.83 Gui::Widgets::ProjectsWidget Class Reference . . . 163
8.83.1 Detailed Description . . . 164
8.83.2 Constructor & Destructor Documentation . . . 164
8.83.2.1 ProjectsWidget . . . 164
8.83.3 Member Function Documentation . . . 164
8.83.3.1 updateBtn . . . 164
8.84 Mustache::QtVariantContext Class Reference . . . 164
8.84.1 Detailed Description . . . 165

8.84.2 Member Typedef Documentation . . . 165
8.84.2.1 fn_t . . . 165
8.84.3 Member Function Documentation . . . 165
8.84.3.1 canEval . . . 165
8.84.3.2 eval . . . 165
8.84.3.3 isFalse . . . 166
8.84.3.4 listCount . . . 166
8.84.3.5 pop . . . 166
8.84.3.6 push . . . 166
8.84.3.7 stringValue . . . 166
8.85 Models::Rate Class Reference . . . 166
8.85.1 Detailed Description . . . 167
8.85.2 Constructor & Destructor Documentation . . . 167
8.85.2.1 Rate . . . 167
8.85.2.2 Rate . . . 167
8.85.3 Member Function Documentation . . . 167
8.85.3.1 getDailyRate . . . 167
8.85.3.2 getHourlyRate . . . 168
8.85.3.3 getLegalRate . . . 168
8.85.3.4 getNbDailyHours . . . 168
8.86 Databases::RateDatabase Class Reference . . . 168
8.86.1 Member Function Documentation . . . 169
8.86.1.1 addRateProject . . . 169
8.86.1.2 getRate . . . 169
8.86.1.3 instance . . . 169
8.86.1.4 updateRateProject . . . 169
8.87 RateModelTest Class Reference . . . 170
8.88 Gui::Widgets::RateWidget Class Reference . . . 170
8.88.1 Detailed Description . . . 171
8.88.2 Constructor & Destructor Documentation . . . 171
8.88.2.1 RateWidget . . . 171
8.88.3 Member Function Documentation . . . 171
8.88.3.1 getDailyRate . . . 171
8.88.3.2 getHourlyRate . . . 171
8.88.3.3 setDailyRate . . . 171
8.88.3.4 setHourlyRate . . . 171
8.88.3.5 setWidgetDailyRateValue . . . 171
8.88.3.6 setWidgetHourlyRateValue . . . 172
8.89 Mustache::Renderer Class Reference . . . 172
8.89.1 Detailed Description . . . 172

8.89.2 Member Function Documentation . . . 172
8.89.2.1 error . . . 172
8.89.2.2 errorPartial . . . 172
8.89.2.3 errorPos . . . 172
8.89.2.4 render . . . 173
8.89.2.5 setTagMarkers . . . 173
8.90 Models::Search Class Reference . . . 173
8.90.1 Detailed Description . . . 174
8.90.2 Member Function Documentation . . . 174
8.90.2.1 filterOnBillsOrQuotes . . . 174
8.90.2.2 filterOnCompany . . . 174
8.90.2.3 filterOnContributories . . . 174
8.90.2.4 filterOnCustomersWithoutProject . . . 175
8.90.2.5 filterOnNumberElements . . . 175
8.90.2.6 filterOnProjects . . . 175
8.90.2.7 filterOnReferentLastname . . . 175
8.90.2.8 filterOnVarcharElements . . . 175
8.90.2.9 getFilter . . . 176
8.90.2.10 getGroupFilter . . . 176
8.90.2.11 getSearchInBillsQuotes . . . 176
8.90.2.12 getSearchInCompanies . . . 176
8.90.2.13 getSearchInProjects . . . 176
8.90.2.14 getSearchInReferentLastname . . . 176
8.90.2.15 getText . . . 176
8.90.2.16 searchInContributories . . . 177
8.90.2.17 setGroupFilter . . . 177
8.90.2.18 setSearchInBillsQuotes . . . 177
8.90.2.19 setSearchInCompanies . . . 177
8.90.2.20 setSearchInContributories . . . 177
8.90.2.21 setSearchInProjects . . . 177
8.90.2.22 setSearchInReferentLastname . . . 177
8.90.2.23 setText . . . 178
8.91 Gui::Docks::SearchDock Class Reference . . . 178
8.91.1 Detailed Description . . . 178
8.91.2 Constructor & Destructor Documentation . . . 179
8.91.2.1 SearchDock . . . 179
8.91.3 Member Function Documentation . . . 180
8.91.3.1 search . . . 180
8.91.3.2 textChanged . . . 180
8.92 searchTest Class Reference . . . 180

8.93 Gui::Widgets::searchWidget Class Reference . . . 180
8.93.1 Detailed Description . . . 181
8.93.2 Constructor & Destructor Documentation . . . 181
8.93.2.1 searchWidget . . . 181
8.93.3 Member Function Documentation . . . 182
8.93.3.1 getCurrentCustomerId . . . 182
8.93.3.2 getIdCustomer . . . 182
8.93.3.3 isCustomerSelected . . . 182
8.93.3.4 search . . . 182
8.93.3.5 selectCustomer . . . 182
8.93.3.6 setIdCustomer . . . 182
8.94 Gui::Dialogs::StartedWindowsDialog Class Reference . . . 183
8.94.1 Detailed Description . . . 184
8.94.2 Constructor & Destructor Documentation . . . 184
8.94.2.1 StartedWindowsDialog . . . 184
8.94.3 Member Function Documentation . . . 184
8.94.3.1 databaseTypeChanged . . . 184
8.94.3.2 getImage . . . 184
8.94.3.3 isDatabaseCentralized . . . 185
8.94.3.4 isDatabaseTypeValid . . . 185
8.95 Models::Statistics Class Reference . . . 185
8.95.1 Member Function Documentation . . . 185
8.95.1.1 getTurnoverBetweenDates . . . 185
8.96 Utils::String Class Reference . . . 185
8.96.1 Detailed Description . . . 186
8.96.2 Member Function Documentation . . . 186
8.96.2.1 firstLetterToUpper . . . 186
8.97 StringTest Class Reference . . . 186
8.98 Mustache::Tag Struct Reference . . . 186
8.98.1 Detailed Description . . . 187
8.98.2 Member Enumeration Documentation . . . 187
8.98.2.1 Type . . . 187
8.99 TestAdder< T > Class Template Reference . . . 187
8.100testadder Class Reference . . . 187
8.101TestMustache Class Reference . . . 188
8.102TestRunner Class Reference . . . 188
8.103Generator::TexGenerator Class Reference . . . 188
8.103.1 Detailed Description . . . 188
8.103.2 Constructor & Destructor Documentation . . . 189
8.103.2.1 TexGenerator . . . 189

8.103.3 Member Function Documentation . . . 190
8.103.3.1 generate . . . 190
8.104 Gui::Widgets::Delegates::TextareaDelegate Class Reference . . . 190
8.104.1 Detailed Description . . . 190
8.104.2 Constructor & Destructor Documentation . . . 191
8.104.2.1 TextareaDelegate . . . 191
8.104.3 Member Function Documentation . . . 191
8.104.3.1 createEditor . . . 191
8.104.3.2 setEditorData . . . 191
8.104.3.3 setModelData . . . 191
8.104.3.4 updateEditorGeometry . . . 192
8.105 Models::Unit Class Reference . . . 193
8.105.1 Detailed Description . . . 193
8.105.2 Constructor & Destructor Documentation . . . 193
8.105.2.1 Unit . . . 193
8.105.3 Member Function Documentation . . . 193
8.105.3.1 getype . . . 193
8.105.3.2 operator"!= . . . 194
8.105.3.3 operator== . . . 194
8.105.3.4 setType . . . 194
8.105.3.5 toString . . . 194
8.106 Gui::Widgets::Delegates::UnitComboDelegate Class Reference . . . 194
8.106.1 Detailed Description . . . 195
8.106.2 Constructor & Destructor Documentation . . . 195
8.106.2.1 UnitComboDelegate . . . 195
8.106.3 Member Function Documentation . . . 195
8.106.3.1 createEditor . . . 195
8.106.3.2 paint . . . 196
8.107 Models::User Class Reference . . . 196
8.107.1 Detailed Description . . . 197
8.107.2 Constructor & Destructor Documentation . . . 197
8.107.2.1 User . . . 197
8.107.3 Member Function Documentation . . . 197
8.107.3.1 getDataMap . . . 197
8.107.3.2 getNbDaysPerMonths . . . 198
8.107.3.3 getNbHoursPerDays . . . 198
8.107.3.4 getNoSiret . . . 198
8.107.3.5 getPdflatexPath . . . 198
8.107.3.6 getTitle . . . 198
8.107.3.7 getWorkspaceName . . . 198

8.107.3.8 getWorkspacePath . . . 199
8.107.3.9 hydrat . . . 199
8.107.3.10 operator"!= . . . 199
8.107.3.11 operator== . . . 199
8.107.3.12 setNoSiret . . . 199
8.107.3.13 setPdflatexPath . . . 199
8.107.3.14 setTitle . . . 200
8.107.3.15 setWorkspaceName . . . 200
8.107.3.16 setWorkspacePath . . . 200
8.107.3.17 updateFolders . . . 200
8.108 Databases::UserDatabase Class Reference . . . 200
8.108.1 Detailed Description . . . 201
8.108.2 Member Function Documentation . . . 201
8.108.2.1 getUser . . . 201
8.108.2.2 instance . . . 201
8.109 UserDatabaseTest Class Reference . . . 202
8.110 Gui::Dialogs::UserDataDialog Class Reference . . . 202
8.110.1 Detailed Description . . . 203
8.110.2 Constructor & Destructor Documentation . . . 203
8.110.2.1 UserDataDialog . . . 203
8.111 UserModelTest Class Reference . . . 203

Index **205**

Chapter 1

FactDev documentation

This website contains the documentation of FactDev, a billings and quotes software.

Team

FACT team is a team for Universitiy Project. Members of this team are :

- Florent Berbie (KraTuX31)
- Antoine de Roquemaurel (aroquemaurel)
- Cédric Rohaut (Oxynos)
- Manantsoa Andriamihary Razanajatovo (manantsoa)

For more informations, you can go to http://fact-team.github.io.

What is FactDev ?

FactDev is a software for quotes and billings developped by FACT team for university project in Toulouse II↩ I University – Paul Sabatier.

This software is developped with C++ and Qt framework, and there is differents features :

- Customers database
- Differents projects for customers
- Quotes
- Billings

Documentation

You can access to the documentation generated by Doxygen here :

- HTML Documentation
- PDF Documentation

Installation and using

- User Manual

Chapter 2

README

`![Build Status](https://travis-ci.org/robertknight/qt-mustache.svg?branch=master)`

Qt Mustache

qt-mustache is a simple library for rendering `Mustache templates`.

Example Usage

“‘cpp #include "mustache.h"

QVariantHash contact; contact["name"] = "John Smith"; contact["email"] = "john.smith@gmail.com";

QString contactTemplate = "{{name}} <a href=\"mailto:{{email}}">{{email}}";

Mustache::Renderer renderer; Mustache::QtVariantContext context(contact);

QTextStream output(stdout); output << renderer.render(contactTemplate, &context); “‘

Outputs: `<b>John Smith</b> <a href="mailto:john.smith\@gmail.com">john.↩ smith@gmail.com</a>`

For further examples, see the tests in `test_mustache.cpp`

Building

- To build the tests, run `qmake` followed by `make`
- To use qt-mustache in your project, just add the `mustache.h` and `mustache.cpp` files to your project.

License

qt-mustache is licensed under the BSD license.

Dependencies

qt-mustache depends on the QtCore library. It is compatible with Qt 4 and Qt 5.

Usage

Syntax

qt-mustache uses the standard Mustache syntax. See the Mustache manual for details.

Data Sources

qt-mustache expands Mustache tags using values from a Mustache::Context. Mustache::QtVariant↩ Context is a simple context implementation which wraps a QVariantHash or QVariantMap. If you want to render a template using a custom data source, you can either create a QVariantHash which mirrors the data source or you can re-implement Mustache::Context.

Partials

When a {{>partial}} Mustache tag is encountered, qt-mustache will attempt to load the partial using a Mustache::PartialResolver provided by the context. Mustache::PartialMap is a simple resolver which takes a QHash<QString,QString> map of partial names to values and looks up partials in that map. Mustache::PartialFileLoader is another simple resolver which fetches partials from <partial name>.mustache files in a specified directory.

You can re-implement the Mustache::PartialResolver interface if you want to load partials from a custom source (eg. a database).

Error Handling

If an error occurs when rendering a template, Mustache::Renderer::errorPosition() is set to non-negative value and template rendering stops. If the error occurs whilst rendering a partial template, error↩ Partial() contains the name of the partial.

Lambdas

The Mustache manual provides a mechanism to customize rendering of template sections by setting the value for a tag to a callable object (eg. a lambda in Ruby or Javascript), which takes the unrendered block of text for a template section and renders it itself. qt-mustache supports this via the Context::canEval() and Context↩ ::eval() methods.

Chapter 3

QTestRunner

A repo for Increase QTest productivity

Thanks to https://marcoarena.wordpress.com/2012/06/23/increase-your-qtest-productivity/
!

> The QTestLib framework is a tool for unit testing Qt based applications and libraries. I find it precious and simple to use, though it lacks some important features, supported, for example, by GTest. I'm not talking about mocking – for this you generally need an out-and-out framework (like GMock) – instead, I'm referring to simple things like fast deploying. Suppose you have written a test class like this:

This repo contains only the code in this article.

Chapter 4

Namespace Index

4.1 Namespace List

Here is a list of all documented namespaces with brief descriptions:

Databases
    Class for Database, contains queries . . . . . ??
Gui . . . . . ??
Gui::Widgets
    Contains Widgets classes . . . . . ??
Models
    Models classes . . . . . ??

Chapter 5

Hierarchical Index

5.1 Class Hierarchy

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

Databases::AccessDatabase . . . 21
Models::Calculable . . . 42
  Models::Billing . . . 25
  Models::ContributoriesList . . . 73
  Models::Contributory . . . 84
Mustache::Context . . . 71
  Mustache::QtVariantContext . . . 164
    CounterContext . . . 92
    CounterContext . . . 92
Databases::Database . . . 106
  Databases::BillingDatabase . . . 31
  Databases::ContributoryDatabase . . . 88
  Databases::CustomerDatabase . . . 96
  Databases::ProjectDatabase . . . 155
  Databases::RateDatabase . . . 168
  Databases::UserDatabase . . . 200
Utils::Directories . . . 114
Utils::Double . . . 114
exception
  Exceptions::DbException . . . 111
Exceptions::FileException . . . 117
Utils::HierarchicalSystem . . . 119
Gui::Widgets::CheckFields::ICheckField . . . 121
  Gui::Widgets::CheckFields::CheckQLineEdit . . . 56
    Gui::Widgets::CheckFields::CheckEmail . . . 44
    Gui::Widgets::CheckFields::CheckPortNumber . . . 53
    Gui::Widgets::CheckFields::CheckSiretNumber . . . 57
    Gui::Widgets::CheckFields::CheckUntilField . . . 58
      Gui::Widgets::CheckFields::CheckFieldsLetters . . . 46
        Gui::Widgets::CheckFields::CheckCity . . . 43
        Gui::Widgets::CheckFields::CheckCountry . . . 43
        Gui::Widgets::CheckFields::CheckName . . . 51
      Gui::Widgets::CheckFields::CheckFieldsNumbers . . . 47
      Gui::Widgets::CheckFields::CheckIpAddress . . . 48
      Gui::Widgets::CheckFields::CheckLogin . . . 49
      Gui::Widgets::CheckFields::CheckPhone . . . 51
      Gui::Widgets::CheckFields::CheckPostalCode . . . 54

Gui::Widgets::CheckFields::CheckValidField . . . 59
Gui::Widgets::CheckFields::CheckWebsite . . . 61
Models::IModel . . . 122
Models::Billing . . . 25
Models::Contributory . . . 84
Models::Customer . . . 93
Models::Project . . . 144
Models::User . . . 196
Utils::ItemType . . . 124
Utils::Log . . . 126
Parameters . . . 133
Mustache::PartialResolver . . . 135
Mustache::PartialFileLoader . . . 134
Mustache::PartialMap . . . 134
Generator::PdfGenerator . . . 136
Models::People . . . 137
Models::Customer . . . 93
Models::User . . . 196
Utils::pointers . . . 143
Gui::Widgets::Popup . . . 143
QAbstractTableModel
Gui::Widgets::WdgModels::BillingsTableModel . . . 38
Gui::Widgets::WdgModels::ContributoriesTableModel . . . 78
Gui::Widgets::WdgModels::CustomersTableModel . . . 102
Gui::Widgets::WdgModels::ProjectContributoriesTableModel . . . 151
Gui::Widgets::WdgModels::ProjectsTableModel . . . 160
QDialog
Gui::Dialogs::AddProjectDialog . . . 21
Gui::Dialogs::AddQuoteDialog . . . 22
Gui::Dialogs::ComputeTurnoverDialog . . . 70
Gui::Dialogs::DialogAddCustomer . . . 113
Gui::Dialogs::MessageBox . . . 132
Gui::Dialogs::StartedWindowsDialog . . . 183
Gui::Dialogs::UserDataDialog . . . 202
QDockWidget
Gui::Docks::SearchDock . . . 178
QItemDelegate
Gui::Widgets::Delegates::ComboBoxDelegate . . . 66
Gui::Widgets::Delegates::ProjectComboDelegate . . . 149
Gui::Widgets::Delegates::UnitComboDelegate . . . 194
Gui::Widgets::Delegates::DoubleSpinBoxDelegate . . . 115
Gui::Widgets::Delegates::TextareaDelegate . . . 190
QLineEdit
Gui::Widgets::CheckFields::CheckQLineEdit . . . 56
QMainWindow
Gui::MainWindow . . . 127
QMap
Models::ContributoriesList . . . 73
QMenu
Gui::Widgets::CustomerContextualMenu . . . 95
QObject
BillingDatabaseTest . . . 37
BillingModelTest . . . 37
ContributoriesDatabaseTest . . . 73
ContributoryListTest . . . 91
ContributoryModelTest . . . 91
CustomerDatabaseTest . . . 101

CustomerModelTest . . . 102
Generation . . . 119
ItemTypeTest . . . 126
PointersTest . . . 143
ProjectDatabaseTest . . . 159
ProjectModelTest . . . 160
RateModelTest . . . 170
searchTest . . . 180
StringTest . . . 186
TestMustache . . . 188
TestMustache . . . 188
UserDatabaseTest . . . 202
UserModelTest . . . 203
QWidget
FileChoseWidget . . . 117
Gui::Widgets::ComboBoxModelWidget . . . 69
Gui::Widgets::ContributoriesWidget . . . 82
Gui::Widgets::CustomerDataWidget . . . 101
Gui::Widgets::DatabaseSettingsWidget . . . 109
Gui::Widgets::Path::ChosePathWidget . . . 64
Gui::Widgets::Path::ChoseDirectoryWidget . . . 62
Gui::Widgets::Path::ChoseFileWidget . . . 63
Gui::Widgets::ProjectsWidget . . . 163
Gui::Widgets::RateWidget . . . 170
Gui::Widgets::searchWidget . . . 180
Models::Rate . . . 166
Mustache::Renderer . . . 172
Models::Search . . . 173
Models::Statistics . . . 185
Utils::String . . . 185
Mustache::Tag . . . 186
TestAdder< T > . . . 187
testadder . . . 187
TestRunner . . . 188
Generator::TexGenerator . . . 188
Models::Unit . . . 193

Chapter 6

Class Index

6.1 Class List

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

Databases::AccessDatabase . . . . . ??

Gui::Dialogs::AddProjectDialog
Windows to add a new Project . . . . . ??

Gui::Dialogs::AddQuoteDialog
Window to add or modify a Quote . . . . . ??

Models::Billing
: Billing or Quote of a Customer . . . . . ??

Databases::BillingDatabase
The **BillingDatabase** class Billing (or Quote) table database . . . . . ??

BillingDatabaseTest . . . . . ??

BillingModelTest . . . . . ??

Gui::Widgets::WdgModels::BillingsTableModel
For a Billing table . . . . . ??

Models::Calculable
The Calculable interface Models who are calculable . . . . . ??

Gui::Widgets::CheckFields::CheckCity
Line Edit of City with a check icon . . . . . ??

Gui::Widgets::CheckFields::CheckCountry
CheckCountry::CheckCountry Line Edit of country with a check icon . . . . . ??

Gui::Widgets::CheckFields::CheckEmail
Line Edit of email with a check icon . . . . . ??

Gui::Widgets::CheckFields::CheckFieldsLetters
Field with only letters (no numbers) . . . . . ??

Gui::Widgets::CheckFields::CheckFieldsNumbers
Line Edit of number with a check icon . . . . . ??

Gui::Widgets::CheckFields::CheckIpAddress
Line Edit of IP address with a check icon . . . . . ??

Gui::Widgets::CheckFields::CheckLogin
Line Edit of login with a check icon . . . . . ??

Gui::Widgets::CheckFields::CheckName
Line edit of name with a check icon . . . . . ??

Gui::Widgets::CheckFields::CheckPhone
Line Edit of Phone number with a check icon . . . . . ??

Gui::Widgets::CheckFields::CheckPortNumber
The CheckFieldsNumbers class Line Edit of number with a check icon . . . . . ??

Gui::Widgets::CheckFields::CheckPostalCode
Line Edit of postal code with a check icon . . . . . ??

Gui::Widgets::CheckFields::CheckQLineEdit
LineEdit custom with a check of text inputed . . . ??
Gui::Widgets::CheckFields::CheckSiretNumber
Line Edit with a check icon . . . ??
Gui::Widgets::CheckFields::CheckUntilField
The CheckUntilField class . . . ??
Gui::Widgets::CheckFields::CheckValidField
Check field not required . . . ??
Gui::Widgets::CheckFields::CheckWebsite
Line Edit of website with a check icon . . . ??
Gui::Widgets::Path::ChoseDirectoryWidget
Open a QFileDialog.and display path in textfield . . . ??
Gui::Widgets::Path::ChoseFileWidget
Chose a File in computer . . . ??
Gui::Widgets::Path::ChosePathWidget
The ChoseDirectoryWidget class Open a QFileDialog.and display path in textfield . . . ??
Gui::Widgets::Delegates::ComboBoxDelegate
The ComboBoxDelegate class . . . ??
Gui::Widgets::ComboBoxModelWidget
Model of ComboBox . . . ??
Gui::Dialogs::ComputeTurnoverDialog
Window to compute a turnover with a period . . . ??
Mustache::Context . . . ??
ContributoriesDatabaseTest . . . ??
Models::ContributoriesList
List of contributories . . . ??
Gui::Widgets::WdgModels::ContributoriesTableModel
For a custom table for contributories widget . . . ??
Gui::Widgets::ContributoriesWidget
Widget of Contributories . . . ??
Models::Contributory
The Unit enum Unity of work : hour or day . . . ??
Databases::ContributoryDatabase
The **ContributoryDatabase** class Contributory (or Quote) table database . . . ??
ContributoryListTest . . . ??
ContributoryModelTest . . . ??
CounterContext . . . ??
Models::Customer
Customer . . . ??
Gui::Widgets::CustomerContextualMenu
Display contextual menu on a customer . . . ??
Databases::CustomerDatabase
The **CustomerDatabase** class Customer table database . . . ??
CustomerDatabaseTest . . . ??
Gui::Widgets::CustomerDataWidget
Class for display info of a customer . . . ??
CustomerModelTest . . . ??
Gui::Widgets::WdgModels::CustomersTableModel
For a customer table . . . ??
Databases::Database
The **Database** class Master class for all database access . . . ??
Gui::Widgets::DatabaseSettingsWidget
Windows of database settings . . . ??
Exceptions::DbException
For database exception : queries, db file, ... ??
Gui::Dialogs::DialogAddCustomer
Window to add or modify a Customer . . . ??
Utils::Directories . . . ??

Utils::Double
Utils functions for Double calculs . . . ??
Gui::Widgets::Delegates::DoubleSpinBoxDelegate
The DoubleSpinBoxDelegate class . . . ??
FileChoseWidget . . . ??
Exceptions::FileException
For file/acess file exception . . . ??
Generation . . . ??
Utils::HierarchicalSystem
Create class which contains hierarchical system of FactDev . . . ??
Gui::Widgets::CheckFields::ICheckField
Interface to check fields validity . . . ??
Models::IModel
The IModel class . . . ??
Utils::ItemType
Item type model . . . ??
ItemTypeTest . . . ??
Utils::Log
For Simple management of log . . . ??
Gui::MainWindow
Main Window of the software . . . ??
Gui::Dialogs::MessageBox
Information window with message . . . ??
Parameters
Class for simple user parameters . . . ??
Mustache::PartialFileLoader . . . ??
Mustache::PartialMap . . . ??
Mustache::PartialResolver . . . ??
Generator::PdfGenerator
Generator of PDF files . . . ??
Models::People
People . . . ??
Utils::pointers . . . ??
PointersTest . . . ??
Gui::Widgets::Popup
Class for display popup quickly . . . ??
Models::Project
: Project linked to a Customer . . . ??
Gui::Widgets::Delegates::ProjectComboDelegate
The ProjectComboDelegate class . . . ??
Gui::Widgets::WdgModels::ProjectContributoriesTableModel
Table model of contributories linked to projets . . . ??
Databases::ProjectDatabase
Project table database . . . ??
ProjectDatabaseTest . . . ??
ProjectModelTest . . . ??
Gui::Widgets::WdgModels::ProjectsTableModel
For a Project table . . . ??
Gui::Widgets::ProjectsWidget
Actions on Project . . . ??
Mustache::QtVariantContext . . . ??
Models::Rate
Rate of a prestation . . . ??
Databases::RateDatabase . . . ??
RateModelTest . . . ??
Gui::Widgets::RateWidget
Class for display Rate . . . ??
Mustache::Renderer . . . ??

Models::Search
The Search class . . . ??
Gui::Docks::SearchDock
Dock which contains search bar . . . ??
searchTest . . . ??
Gui::Widgets::searchWidget
Class for search in database . . . ??
Gui::Dialogs::StartedWindowsDialog
Contruct a Windows for the first begin . . . ??
Models::Statistics . . . ??
Utils::String
The Utils class . . . ??
StringTest . . . ??
Mustache::Tag . . . ??
TestAdder< T > . . . ??
testadder . . . ??
TestMustache . . . ??
TestRunner . . . ??
Generator::TexGenerator
Generate a LaTeX file . . . ??
Gui::Widgets::Delegates::TextareaDelegate
The TextareaDelegate class . . . ??
Models::Unit
An unity for billing calculs . . . ??
Gui::Widgets::Delegates::UnitComboDelegate
The UnitComboDelegate class . . . ??
Models::User
User of it application . . . ??
Databases::UserDatabase
Access to User data in the the table User of the **Database** . . . ??
UserDatabaseTest . . . ??
Gui::Dialogs::UserDataDialog
Window to fill user data . . . ??
UserModelTest . . . ??

Chapter 7

Namespace Documentation

7.1 Databases Namespace Reference

Class for Database, contains queries.

Classes

- class AccessDatabase
- class BillingDatabase

  *The **BillingDatabase** class Billing (or Quote) table database.*
- class ContributoryDatabase

  *The **ContributoryDatabase** class Contributory (or Quote) table database.*
- class CustomerDatabase

  *The **CustomerDatabase** class Customer table database.*
- class Database

  *The **Database** class Master class for all database access.*
- class ProjectDatabase

  The ProjectDatabase class Project table database.
- class RateDatabase
- class UserDatabase

  *The UserDatabase class Access to User data in the the table User of the **Database***

Enumerations

- enum **DbType** { **SQLITE**, **MYSQL** }

7.1.1 Detailed Description

Class for Database, contains queries.

7.2 Gui Namespace Reference

Namespaces

- Widgets

  Contains Widgets classes.

Classes

- class MainWindow

  The MainWindow class Main Window of the software.

7.2.1 Detailed Description

Classes for Graphical User Interface

7.3 Gui::Widgets Namespace Reference

Contains Widgets classes.

Classes

- class ComboBoxModelWidget

  The ComboBoxModelWidget class Model of ComboBox.
- class ContributoriesWidget

  The ContributoriesWidget class Widget of Contributories.
- class CustomerContextualMenu

  Display contextual menu on a customer.
- class CustomerDataWidget

  Class for display info of a customer.
- class DatabaseSettingsWidget

  The DatabaseSettingsWidget class Windows of database settings.
- class Popup

  Class for display popup quickly.
- class ProjectsWidget

  The ProjectsWidget class Actions on Project.
- class RateWidget

  Class for display Rate.
- class searchWidget

  Class for search in database.

7.3.1 Detailed Description

Contains Widgets classes.

7.4 Models Namespace Reference

Models classes.

Classes

- class Billing

  The Billing class : Billing or Quote of a Customer.
- class Calculable

  The Calculable interface Models who are calculable.

- class ContributoriesList

  The ContributoriesList class List of contributories.
- class Contributory

  The Unit enum Unity of work : hour or day.
- class Customer

  The Customer class Customer.
- class IModel

  The IModel class.
- class People

  The People class People.
- class Project

  The Project class : Project linked to a Customer.
- class Rate

  The Rate class Rate of a prestation.
- class Search

  The Search class.
- class Statistics
- class Unit

  The Unit class An unity for billing calculs.
- class User

  *The User class **User** of it application.*

Enumerations

- enum TypeUnit { **HOUR**, **DAY**, **MONTH** }

  The TypeUnit enum Type Unit : hour or days.

7.4.1 Detailed Description

Models classes.

Chapter 8

Class Documentation

8.1 Databases::AccessDatabase Class Reference

Static Public Member Functions

- static void **init** ()
- static void **commit** ()

Static Public Attributes

- static QString **_address**
- static int **_port**
- static QString **_userDb**
- static QString **_password**
- static QString **_dbName**
- static Databases::DbType **_dbType**
- static bool **_exists**

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

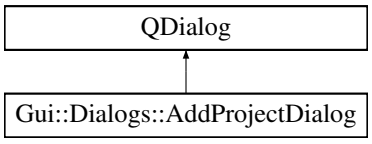
- src/database/accessdatabase.h
- src/database/accessdatabase.cpp

8.2 Gui::Dialogs::AddProjectDialog Class Reference

The AddProjectDialog class Windows to add a new Project.

`#include <addprojectdialog.h>`

Inheritance diagram for Gui::Dialogs::AddProjectDialog:



Public Slots

- void checkFields ()

  AddProjectDialog::checkFields Check if fields are valid.

Public Member Functions

- AddProjectDialog (int idProject=0, int noRowCustomer=0, QWidget ∗parent=0)

  AddProjectDialogAddProjectDialog Construct a windows according an idCustomer and, optionnaly, an idProject
- void accept ()

  AddProjectDialog::accept Valid data inputed by user and add these data in Database.
- void reject ()

  AddProjectDialog::reject Cancel the operation and close the windows.
- void fillFields ()

  AddProjectDialog::fillFields Fill the differents fields of the current windows according the Project data existing As a project requires to be linked to a Customer, the Customer selection part may be disable.

8.2.1 Detailed Description

The AddProjectDialog class Windows to add a new Project.

Author

Florent Berbie

See also

Project

8.2.2 Constructor & Destructor Documentation

8.2.2.1 Gui::Dialogs::AddProjectDialog::AddProjectDialog (int *idProject* = `0`, int *noRowCustomer* = `0`, QWidget ∗ *parent* = `0`) `[explicit]`

AddProjectDialogAddProjectDialog Construct a windows according an *idCustomer* and, optionnaly, an *idProject*

Parameters

noRowCustomer	Row number of the Customer
idProject	Project identify
parent	QWidget of the current windows

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

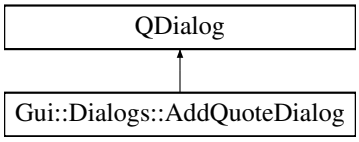
- src/gui/dialogs/addprojectdialog.h
- src/gui/dialogs/addprojectdialog.cpp

8.3 Gui::Dialogs::AddQuoteDialog Class Reference

The AddQuoteDialog class Window to add or modify a Quote.

`#include <addquotedialog.h>`

Inheritance diagram for Gui::Dialogs::AddQuoteDialog:



Public Slots

- void **updateBtn** (void)
- void **changeDocType** ()

Public Member Functions

- AddQuoteDialog (bool isBilling, int idCustomer=0, int id=0, bool copy=false, QWidget ∗parent=0)

  AddQuoteDialog::AddQuoteDialog Construct a windows AddQuoteDialog.
- void fillFields ()

  AddQuoteDialog::Fill line edits with the data of the quote.
- int getNumber ()

  AddQuoteDialog::getNumber return the number of bill or quote.
- void accept ()

  AddQuoteDialog::accept Valid data inputed by user and add these data in Database.
- bool getCopy () const

  *AddQuoteDialog::getCopy return if **AddQuoteDialog** is an add by copy of Billing/quote or if it's edition or add of Billing/quote.*
- void setCopy (bool copy)

  AddQuoteDialog::setCopy Change the _copy value to define if it's a copy of a Billing/quote or if it's a new Billing or a Billing edition.
- int getIdCustomer () const

  AddQuoteDialog::getIdCustomer get the id of the customer changing his quote/billing.
- void fillQuoteBilling (bool isBilling)

  AddQuoteDialog::fillQuoteBilling fill fields for a Billing or a quote.
- void fillQuoteBillingCopy (bool isBilling)

  AddQuoteDialog::fillQuoteBilling fill fields if it's a Billing or a quote for a copy.
- void setQuoteIdNumber (int id, int idCustomer, bool isBilling)

  AddQuoteDialog::setQuoteIdNumber set id and number of the _quote

8.3.1 Detailed Description

The AddQuoteDialog class Window to add or modify a Quote.

Author

8.3.2 Constructor & Destructor Documentation

8.3.2.1 Gui::Dialogs::AddQuoteDialog::AddQuoteDialog (bool *isBilling*, int *idCustomer* = `0`, int *id* = `0`, bool *copy* = `false`, QWidget ∗ *parent* = `0`) `[explicit]`

AddQuoteDialog::AddQuoteDialog Construct a windows AddQuoteDialog.

Parameters

isBilling	displaying of a quote or a Billing
idCustomer	the id of the Customer
id	the id of the quote or the billing of the Customer's Project
edit	if it's an edition or a copy of a quote/Billing
parent	QWidget of the current windows

8.3.3 Member Function Documentation

8.3.3.1 void Gui::Dialogs::AddQuoteDialog::fillQuoteBilling (bool *isBilling*)

AddQuoteDialog::fillQuoteBilling fill fields for a Billing or a quote.

Parameters

isBilling	if it's a quote or a Billing

8.3.3.2 bool Gui::Dialogs::AddQuoteDialog::getCopy ( ) const

AddQuoteDialog::getCopy return if **AddQuoteDialog** is an add by copy of Billing/quote or if it's edition or add of Billing/quote.

Returns

if it's a copy or not

8.3.3.3 int Gui::Dialogs::AddQuoteDialog::getIdCustomer ( ) const

AddQuoteDialog::getIdCustomer get the id of the customer changing his quote/billing.

Returns

the id of the customer

8.3.3.4 int Gui::Dialogs::AddQuoteDialog::getNumber ( )

AddQuoteDialog::getNumber return the number of bill or quote.

Returns

int

8.3.3.5 void Gui::Dialogs::AddQuoteDialog::setCopy (bool *copy*)

AddQuoteDialog::setCopy Change the *_copy* value to define if it's a copy of a Billing/quote or if it's a new Billing or a Billing edition.

Parameters

copy	

8.3.3.6 void Gui::Dialogs::AddQuoteDialog::setQuoteIdNumber (int *id,* int *idCustomer,* bool *isBilling*)

AddQuoteDialog::setQuoteIdNumber set id and number of the *_quote*

Parameters

id	the id of the Billing
idCustomer	the id of the customer to set the *number*
isBilling	if it's a quote or a Billing

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

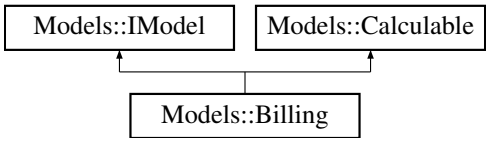
- src/gui/dialogs/addquotedialog.h
- src/gui/dialogs/addquotedialog.cpp

8.4 Models::Billing Class Reference

The Billing class : Billing or Quote of a Customer.

`#include <billing.h>`

Inheritance diagram for Models::Billing:



Public Member Functions

- Billing ()

  Billing::Billing. Construct a Billing.
- Billing (int id)

  Billing::Billing. Construct a Billing or quote.
- ~Billing ()

  destruct a billing object
- void commit ()

  Billing::commit. Insert a modification in Billing table on the database.
- void hydrat (int getId)

  Billing::hydrat. Update of the Billing which is specified by getId
- void remove ()

  Billing::remove. Remove a Billing.
- QVariantHash getDataMap ()

  Billing::getDataMap Get all data of model with a HashMap key/value.
- double getPrice (bool paied=false)

  getPrice Return the price of a calculable object
- double getSumQuantity ()

  ContributoriesList::getSumQuantity Return the sum of quantity (number of days) of the Contributories.
- void generateTex ()

Billing::generateTex *Generate a .tex file for the billing.*

- void generatePdf ()

  Billing::generatePdf *Generate a .pdf file for the billing.*

- QString getPath ()

  Billing::getPath *Return the path of billing filename (without extension)*

- QString getFolder ()

  Billing::getFolder *Return the directory of billing.*

- QString getFilename ()

  Billing::getFilename *Return the filename of billing (without extension)*

- ContributoriesList & getContributories ()

  Billing::getContributories. Return a map of ***Contributory*** *for each* ***Project*** *of the* ***Billing***

- void addContributory (Contributory &c)

  Billing::addContributories Add a new contributory for project p.

- QString getTitle () const

  Billing::getTitle. return title of ***Billing***

- void setTitle (const QString &getTitle)

  Billing::setTitle. Modify the title of ***Billing***

- QString getDescription () const

  Billing::getDescription. return description of ***Billing***

- void setDescription (const QString &getDescription)

  Billing::setDescription. Modify the description of ***Billing***

- int getNumber () const

  Billing::getNumber. Return number of the ***Billing****.*

- void setNumber (int getNumber)

  Billing::setNumber. Modify _number of Billing.

- bool isBilling () const

  Billing::isBilling. Return if it's a billing or a quote.

- void setIsBilling (bool isBilling)

  Billing::setIsBilling. Modify isBilling of ***Billing****.*

- QDate getDate () const

  Billing::getDate. return date of the ***Billing***

- void setDate (const QDate &getDate)

  Billing::setDate. Modify date of the ***Billing***

- bool isPaid () const

  Billing::isPaid *Return TRUE if thee current billing is paid else return FALSE.*

- void setIsPaid (bool isPaid)

  Billing::setIsPaid *Define the current billing according the argument isPaid*

- bool operator== (const Billing &b)

  Billing::operator == *define the operator "==" to compare two billings and to see if they are the same.*

- bool operator!= (const Billing &b)

  Billing::operator != *defines the operator "!=" to compare two* ***Billing*** *and to see if they are different.*

- void **setContributories** (const ContributoriesList &contributories)

- bool operator< (const Billing &b) const

  Billing::operator < *defines the operator "< to compare two* ***Billing*** *and to see if the fisrt is anterior to the second.*

- QStandardItem * getItem ()

  Billing::getItem *Return the bill/quote item.*

Additional Inherited Members

8.4.1 Detailed Description

The Billing class : Billing or Quote of a Customer.

Author

Antoine de Roquemaurel
Florent Berbie

8.4.2 Constructor & Destructor Documentation

8.4.2.1 Models::Billing::Billing (int *id*)

Billing::Billing. Construct a Billing or quote.

Parameters

int	id

8.4.3 Member Function Documentation

8.4.3.1 void Models::Billing::addContributory (Contributory & *c*)

Billing::addContributories Add a new contributory for project p.

Parameters

p	The Project who contain Contributory
c	The new Contributory

8.4.3.2 ContributoriesList & Models::Billing::getContributories ( )

Billing::getContributories. Return a map of **Contributory** for each **Project** of the **Billing**

Returns

QMap<Project, QList<Contributory>>

8.4.3.3 QVariantHash Models::Billing::getDataMap ( ) [virtual]

Billing::getDataMap Get all data of model with a HashMap key/value.

Returns

Model's data

Implements Models::IModel.

8.4.3.4 QDate Models::Billing::getDate ( ) const

Billing::getDate. return date of the **Billing**

Returns

date of Billing

8.4.3.5 QString Models::Billing::getDescription ( ) const

Billing::getDescription. return description of **Billing**

Returns

description of Billing

8.4.3.6 QString Models::Billing::getFilename ( )

Billing::getFilename Return the filename of billing (without extension)

Returns

Filename of Bulling

8.4.3.7 QString Models::Billing::getFolder ( )

Billing::getFolder Return the directory of billing.

Returns

Billing directory

8.4.3.8 QStandardItem * Models::Billing::getItem ( )

Billing::getItem Return the bill/quote item.

Returns

QStandardItem an item for QTree (level/depth 3)

8.4.3.9 int Models::Billing::getNumber ( ) const

Billing::getNumber. Return number of the **Billing**.

Returns

_number of Billing

8.4.3.10 QString Models::Billing::getPath ( )

Billing::getPath Return the path of billing filename (without extension)

Returns

billing path

8.4.3.11 double Models::Billing::getPrice (bool *paied* = `false`) [virtual]

getPrice Return the price of a calculable object

Returns

The price

Implements Models::Calculable.

8.4.3.12 double Models::Billing::getSumQuantity ( ) [virtual]

ContributoriesList::getSumQuantity Return the sum of quantity (number of days) of the Contributories.

Returns

sum of quantity in days

Implements Models::Calculable.

8.4.3.13 QString Models::Billing::getTitle ( ) const

Billing::getTitle. return title of **Billing**

Returns

title of Billing

8.4.3.14 void Models::Billing::hydrat (int *getId*) [virtual]

Billing::hydrat. Update of the Billing which is specified by *getId*

Parameters

getId	

Implements Models::IModel.

8.4.3.15 bool Models::Billing::isBilling ( ) const

Billing::isBilling. Return if it's a billing or a quote.

Returns

if it's billing or a quote

8.4.3.16 bool Models::Billing::isPaid ( ) const

Billing::isPaid Return TRUE if thee current billing is paid else return FALSE.

Returns

Boolean

8.4.3.17 bool Models::Billing::operator!= (const Billing & *b*)

Billing::operator != defines the operator "!=" to compare two **Billing** and to see if they are different.

Parameters

b	the **Billing** to compare with the current **Billing**

Returns

true if the **Billing** are different else false

8.4.3.18 bool Models::Billing::operator< (const Billing & *b*) const

Billing::operator < defines the operator "< to compare two **Billing** and to see if the fisrt is anterior to the second.

Parameters

b	the **Billing** to compare with the current **Billing**

Returns

true if the **Billing** are different else false

8.4.3.19 bool Models::Billing::operator== (const Billing & *b*)

Billing::operator == define the operator "==" to compare two billings and to see if they are the same.

Parameters

b	the **Billing** to compare with the current **Billing**

Returns

true if they are the same billings else false

8.4.3.20 void Models::Billing::setDate (const QDate & *getDate*)

Billing::setDate. Modify *date* of the **Billing**

Parameters

getDate	the new date of the Billing

8.4.3.21 void Models::Billing::setDescription (const QString & *getDescription*)

Billing::setDescription. Modify the description of **Billing**

Parameters

getDescription	Modify the description with *getDescription*

8.4.3.22 void Models::Billing::setIsBilling (bool *isBilling*)

Billing::setIsBilling. Modify *isBilling* of **Billing**.

Parameters

isBilling	

8.4.3.23 void Models::Billing::setIsPaid (bool *isPaid*)

Billing::setIsPaid Define the current billing according the argument *isPaid*

Parameters

isPaid	Boolean

8.4.3.24 void Models::Billing::setNumber (int *getNumber*)

Billing::setNumber. Modify *_number* of Billing.

Parameters

getNumber	the new number of the Billing

8.4.3.25 void Models::Billing::setTitle (const QString & *getTitle*)

Billing::setTitle. Modify the title of **Billing**

Parameters

getTitle	Modify the title with *getTitle*

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

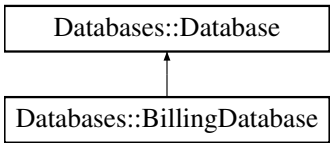
- src/models/billing.h
- src/models/billing.cpp

8.5 Databases::BillingDatabase Class Reference

The **BillingDatabase** class Billing (or Quote) table database.

`#include <billingdatabase.h>`

Inheritance diagram for Databases::BillingDatabase:



Public Member Functions

- Models::Billing * getBilling (const int pId)

  BillingDatabase::getCustomer get informations about the billing identified by pId
- WdgModels::BillingsTableModel * getBillingsTable (const int idProject) throw (DbException*)

  BillingDatabase::getBillingsTable Return an item model of billings for QTableView.
- int addBilling (const Models::Billing &)

BillingDatabase::addBilling Add the billing pBilling to the database.

- void updateBilling (const Models::Billing &)

  BillingDatabase::updateCustomer Update informations about the billing pCustomer

- void removeBilling (const int pId)

  BillingDatabase::removeCustomer Remove the billing with the id pId

- void addBillingProject (const int idProject, const int idBilling, const int idContributory)

  BillingDatabase::addBillingProject Link a project, a billing and a contributory in the table BillingProject.

- bool isBillingPaid (const int pId)

  BillingDatabase::isBillingPaid Return TRUE if the id pId correspond to a Billing and not quote (isBilling = 1) and if this billing is paid (isPaid = 1) else return FALSE.

- void removeBillingProject (const int idProject, const int idBilling, const int idContributory)

  BillingDatabase::removeBillingProject remove a link between a project, a billing and a contributory in the table BillingProject.

- int getMaxBillingNumber ()

  getMaxBillingNumber Get the last number of a billing

- int getMaxQuoteNumber ()

  getMaxQuoteNumber Get the last number of a quote

- int getMaxBillingNumberOfCustomer (const int idCustomer)

  *BillingDatabase::getMaxBillingNumberOfCustomer Get the last number of a **Customer**'s Billing.*

- int getMaxQuoteNumberOfCustomer (const int idCustomer)

  *BillingDatabase::getMaxQuoteNumberOfCustomer Get the last number of a **Customer**'s quote.*

- QSharedPointer< Models::Billing > getBilling (QSqlQuery &q)

  BillingDatabase::getBilling Add the element of the q request and return their.

- QMap< Project *, Billing * > getAllBillingsOfProject ()

  BillingDatabase::getAllBillingsOfProject Return a map with the project id as key linked to the billing.

- QList< Billing > getBillings (const int projectId)

  BillingDatabase::getBillings get bills by project.

- QList< Billing > getAllBillingsOnly (const int idProject)

  BillingDatabase::getAllBillingsOnly get all billings which aren't quotes for the project p

- QList< Billing > getBillingsBetweenDates (QDate begin, QDate end)

  BillingDatabase::getBillingsBetweenDates get billings in the list between begin and end dates.

Static Public Member Functions

- static BillingDatabase * instance () throw (DbException*)

  *BillingDatabase::getInstance Return an instance of **BillingDatabase***

Additional Inherited Members

8.5.1 Detailed Description

The **BillingDatabase** class Billing (or Quote) table database.

Author

See also

Database
Billing/Quote

8.5.2 Member Function Documentation

8.5.2.1 int Databases::BillingDatabase::addBilling (const Models::Billing & *pBilling*)

BillingDatabase::addBilling Add the billing *pBilling* to the database.

Returns

billing id

8.5.2.2 void Databases::BillingDatabase::addBillingProject (const int *idProject*, const int *idBilling*, const int *idContributory*)

BillingDatabase::addBillingProject Link a project, a billing and a contributory in the table BillingProject.

Parameters

idProject	Project id
idBilling	Billing id
idContributory	Contributory id

8.5.2.3 QMap< Project *, Billing * > Databases::BillingDatabase::getAllBillingsOfProject ( )

BillingDatabase::getAllBillingsOfProject Return a map with the project id as key linked to the billing.

Returns

Map with projects and Billing

8.5.2.4 QList< Billing > Databases::BillingDatabase::getAllBillingsOnly (const int *idProject*)

BillingDatabase::getAllBillingsOnly get all billings which aren't quotes for the project *p*

Parameters

p	

Returns

the list of **Billing**

8.5.2.5 Models::Billing * Databases::BillingDatabase::getBilling (const int *pId*)

BillingDatabase::getCustomer get informations about the billing identified by *pId*

Parameters

pId	billing id

Returns

the Billing

8.5.2.6 QSharedPointer< Billing > Databases::BillingDatabase::getBilling (QSqlQuery & *q*)

BillingDatabase::getBilling Add the element of the *q* request and return their.

Parameters

q	SQL request

Returns

a billing formed according to QSharedPointer

8.5.2.7 QList< Billing > Databases::BillingDatabase::getBillings (const int *projectId*)

BillingDatabase::getBillings get bills by project.

Parameters

projectId	

Returns

List with bills

8.5.2.8 QList< Billing > Databases::BillingDatabase::getBillingsBetweenDates (QDate *begin*, QDate *end*)

BillingDatabase::getBillingsBetweenDates get billings in the list between *begin* and *end* dates.

Parameters

begin	
end	

Returns

the list of **Billing**

8.5.2.9 WdgModels::BillingsTableModel * Databases::BillingDatabase::getBillingsTable (const int *idProject*) throw DbException *)

BillingDatabase::getBillingsTable Return an item model of billings for QTableView.

Parameters

pId	the project id of the billings returned

Exceptions

DbException	

Returns

QStandardItemModel an item model for QTableView

8.5.2.10 int Databases::BillingDatabase::getMaxBillingNumber ( )

getMaxBillingNumber Get the last number of a billing

Returns

The max number

8.5.2.11 int Databases::BillingDatabase::getMaxBillingNumberOfCustomer (const int *idCustomer*)

BillingDatabase::getMaxBillingNumberOfCustomer Get the last number of a **Customer**'s Billing.

Parameters

idCustomer	Customer id

Returns

the last number of the **Customer**'s Billing

8.5.2.12 int Databases::BillingDatabase::getMaxQuoteNumber ( )

getMaxQuoteNumber Get the last number of a quote

Returns

The last number

8.5.2.13 int Databases::BillingDatabase::getMaxQuoteNumberOfCustomer (const int *idCustomer*)

BillingDatabase::getMaxQuoteNumberOfCustomer Get the last number of a **Customer**'s quote.

Parameters

idCustomer	Customer id

Returns

the last number of the **Customer**'s quote

8.5.2.14 BillingDatabase * Databases::BillingDatabase::instance ( ) throw DbException *) [static]

BillingDatabase::getInstance Return an instance of **BillingDatabase**

See also

DbException

Returns

Instance of BillingDatabase

8.5.2.15 bool Databases::BillingDatabase::isBillingPaid (const int *pId*)

BillingDatabase::isBillingPaid Return TRUE if the id *pId* correspond to a Billing and not quote (isBilling = 1) and if this billing is paid (isPaid = 1) else return FALSE.

Parameters

pId	Billing id

Returns

TRUE if billing is paid

8.5.2.16 void Databases::BillingDatabase::removeBilling (const int *pId*)

BillingDatabase::removeCustomer Remove the billing with the id *pId*

Parameters

pId	billing id

8.5.2.17 void Databases::BillingDatabase::removeBillingProject (const int *idProject*, const int *idBilling*, const int *idContributory*)

BillingDatabase::removeBillingProject remove a link between a project, a billing and a contributory in the table BillingProject.

Parameters

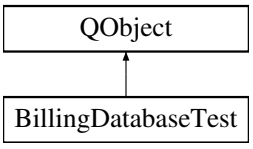
idProject	Project id
idBilling	Billing id
idContributory	Contributory id

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

- src/database/billingdatabase.h
- src/database/billingdatabase.cpp

8.6 BillingDatabaseTest Class Reference

Inheritance diagram for BillingDatabaseTest:

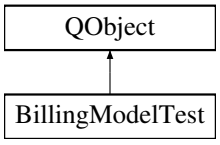


The documentation for this class was generated from the following files:

- tests/database/billingdatabasetest.h
- tests/database/billingdatabasetest.cpp

8.7 BillingModelTest Class Reference

Inheritance diagram for BillingModelTest:



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

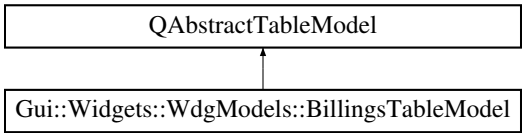
- tests/models/billingmodeltest.h
- tests/models/billingmodeltest.cpp

8.8 Gui::Widgets::WdgModels::BillingsTableModel Class Reference

The BillingsTableModel class for a Billing table.

`#include <billingstablemodel.h>`

Inheritance diagram for Gui::Widgets::WdgModels::BillingsTableModel:



Public Member Functions

- BillingsTableModel ()

  BillingsTableModel::BillingsTableModel Construct a BillingsTableModel.
- int rowCount (const QModelIndex &) const

  BillingsTableModel::rowCount Number of billings row.
- int columnCount (const QModelIndex &) const

  BillingsTableModel::columnCount Number of column of a Billing.
- QVariant data (const QModelIndex &index, int role=Qt::DisplayRole) const

  BillingsTableModel::data Obtains data of a specify cell.
- QVariant headerData (int section, Qt::Orientation orientation, int role=Qt::DisplayRole) const

  BillingsTableModel::headerData Obtains header title of table.
- bool setData (const QModelIndex &index, const QVariant &value, int role=Qt::EditRole)

  BillingsTableModel::setData Change data of a cell.
- void append (const Billing &billing)

  BillingsTableModel::append Add a new line in table.
- void remove (const int i)

  BillingsTableModel::remove Remove a line.
- Qt::ItemFlags flags (const QModelIndex &index) const

  BillingsTableModel::flags Differents table flags.
- int count ()

  BillingsTableModel::count Number of billings in table.
- QList< Billing > getBillings () const

  BillingsTableModel::getbillings Return the list of billings.

8.8.1 Detailed Description

The BillingsTableModel class for a Billing table.

Author

Florent Berbie

See also

Billing

8.8.2 Constructor & Destructor Documentation

8.8.2.1 Gui::Widgets::WdgModels::BillingsTableModel::BillingsTableModel ( )

BillingsTableModel::BillingsTableModel Construct a BillingsTableModel.

Parameters

parent	Parent widget

8.8.3 Member Function Documentation

8.8.3.1 void Gui::Widgets::WdgModels::BillingsTableModel::append (const Billing & *billing*)

BillingsTableModel::append Add a new line in table.

Parameters

Billing	The new Billing

8.8.3.2 int Gui::Widgets::WdgModels::BillingsTableModel::columnCount (const QModelIndex &) const

BillingsTableModel::columnCount Number of column of a Billing.

Returns

The number of column

8.8.3.3 int Gui::Widgets::WdgModels::BillingsTableModel::count ( )

BillingsTableModel::count Number of billings in table.

Returns

The number of billings

8.8.3.4 QVariant Gui::Widgets::WdgModels::BillingsTableModel::data (const QModelIndex & *index*, int *role* = `Qt::DisplayRole`) const

BillingsTableModel::data Obtains data of a specify cell.

Parameters

index	The cell who we want data
role	The role of set

Returns

The data of cell

8.8.3.5 Qt::ItemFlags Gui::Widgets::WdgModels::BillingsTableModel::flags (const QModelIndex & *index*) const

BillingsTableModel::flags Differents table flags.

Parameters

index	The cell who we want to know flags

Returns

Flags

8.8.3.6 QList< Billing > Gui::Widgets::WdgModels::BillingsTableModel::getBillings ( ) const

BillingsTableModel::getbillings Return the list of billings.

Returns

list of billings

8.8.3.7 QVariant Gui::Widgets::WdgModels::BillingsTableModel::headerData (int *section*, Qt::Orientation *orientation*, int *role* = `Qt::DisplayRole`) const

BillingsTableModel::headerData Obtains header title of table.

Parameters

section	The number of column
orientation	The table orientation
role	

Returns

The Title header of column

8.8.3.8 void Gui::Widgets::WdgModels::BillingsTableModel::remove (const int *i*)

BillingsTableModel::remove Remove a line.

Parameters

i	The number of line to remove

8.8.3.9 int Gui::Widgets::WdgModels::BillingsTableModel::rowCount (const QModelIndex &) const

BillingsTableModel::rowCount Number of billings row.

Returns

The number of billings

8.8.3.10 bool Gui::Widgets::WdgModels::BillingsTableModel::setData (const QModelIndex & *index*, const QVariant & *value*, int *role* = `Qt::EditRole`)

BillingsTableModel::setData Change data of a cell.

Parameters

index	The cell to change data
value	The new value
role	The role of cell

Returns

True if we could edit

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

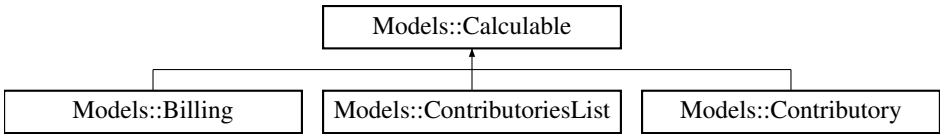
- src/gui/widgets/widgetsmodels/billingstablemodel.h
- src/gui/widgets/widgetsmodels/billingstablemodel.cpp

8.9 Models::Calculable Class Reference

The Calculable interface Models who are calculable.

`#include <calculable.h>`

Inheritance diagram for Models::Calculable:



Public Member Functions

- virtual double getPrice (bool paied=false)=0

  getPrice Return the price of a calculable object
- virtual double getSumQuantity ()=0

  ContributoriesList::getSumQuantity Return the sum of quantity (number of days) of the Contributories.

8.9.1 Detailed Description

The Calculable interface Models who are calculable.

8.9.2 Member Function Documentation

8.9.2.1 virtual double Models::Calculable::getPrice (bool *paied* = `false`) [pure virtual]

getPrice Return the price of a calculable object

Returns

The price

Implemented in Models::Billing, Models::Contributory, and Models::ContributoriesList.

8.9.2.2 virtual double Models::Calculable::getSumQuantity ( ) [pure virtual]

ContributoriesList::getSumQuantity Return the sum of quantity (number of days) of the Contributories.

Returns

sum of quantity in days

Implemented in Models::Billing, Models::Contributory, and Models::ContributoriesList.

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

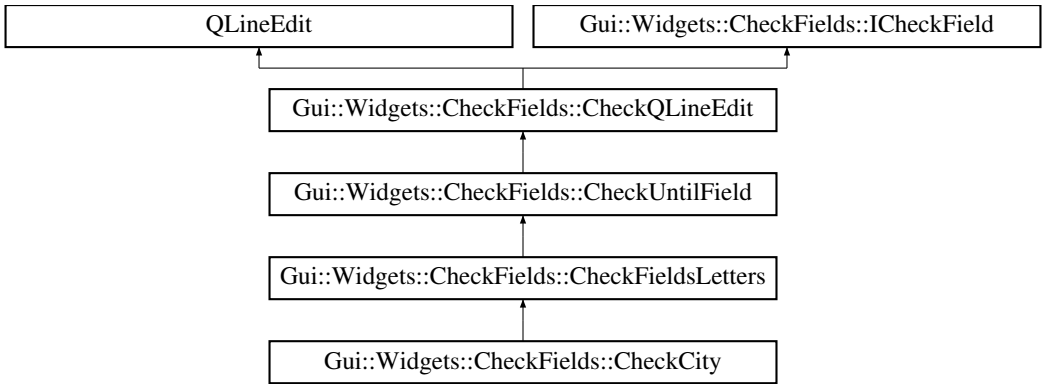
- src/models/calculable.h

8.10 Gui::Widgets::CheckFields::CheckCity Class Reference

The CheckCity class Line Edit of City with a check icon.

`#include <checkcity.h>`

Inheritance diagram for Gui::Widgets::CheckFields::CheckCity:



Public Member Functions

- CheckCity (QWidget *w=0, QPushButton *btn=0)

  CheckCity::CheckCity Construct a CheckCity.

Additional Inherited Members

8.10.1 Detailed Description

The CheckCity class Line Edit of City with a check icon.

8.10.2 Constructor & Destructor Documentation

8.10.2.1 Gui::Widgets::CheckFields::CheckCity::CheckCity (QWidget * *w* = `0`, QPushButton * *btn* = `0`)

CheckCity::CheckCity Construct a CheckCity.

Parameters

w	QWidget linked to **CheckCity**

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

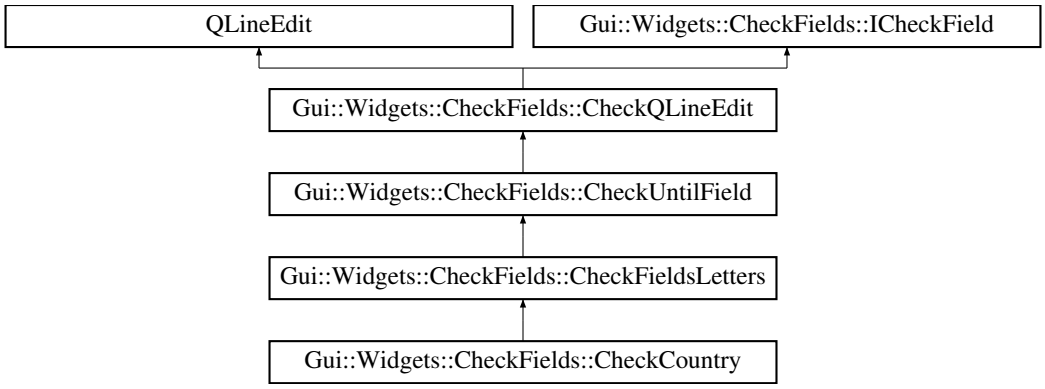
- src/gui/widgets/checkfields/checkcity.h
- src/gui/widgets/checkfields/checkcity.cpp

8.11 Gui::Widgets::CheckFields::CheckCountry Class Reference

CheckCountry::CheckCountry Line Edit of country with a check icon.

`#include <checkcountry.h>`

Inheritance diagram for Gui::Widgets::CheckFields::CheckCountry:



Public Member Functions

- CheckCountry (QWidget ∗w=0, QPushButton ∗btn=0)
  CheckCountry::CheckCountry Construct a CheckCountry.

Additional Inherited Members

8.11.1 Detailed Description

CheckCountry::CheckCountry Line Edit of country with a check icon.

8.11.2 Constructor & Destructor Documentation

8.11.2.1 Gui::Widgets::CheckFields::CheckCountry::CheckCountry (QWidget ∗ *w* = 0, QPushButton ∗ *btn* = 0)

CheckCountry::CheckCountry Construct a CheckCountry.

Parameters

w	QWidget linked to **CheckCountry**

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

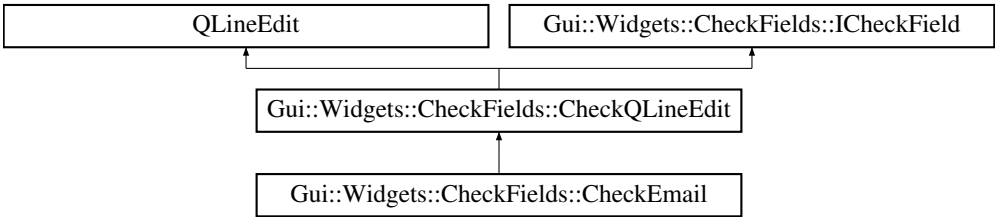
- src/gui/widgets/checkfields/checkcountry.h
- src/gui/widgets/checkfields/checkcountry.cpp

8.12 Gui::Widgets::CheckFields::CheckEmail Class Reference

The CheckEmail class Line Edit of email with a check icon.

`#include <checkemail.h>`

Inheritance diagram for Gui::Widgets::CheckFields::CheckEmail:



Public Member Functions

- CheckEmail (QWidget *w=0, QPushButton *btn=0)

  CheckEmail::CheckEmail Construct a CheckMail.

- bool check (const QString text)

  CheckEmail::check Check if the field email is valid. To be valid, an email address should be under this form : me@me.xx An email address need:

Additional Inherited Members

8.12.1 Detailed Description

The CheckEmail class Line Edit of email with a check icon.

8.12.2 Constructor & Destructor Documentation

8.12.2.1 Gui::Widgets::CheckFields::CheckEmail::CheckEmail (QWidget * *w* = 0, QPushButton * *btn* = 0)

CheckEmail::CheckEmail Construct a CheckMail.

Parameters

w	QWidget linked to **CheckEmail**

8.12.3 Member Function Documentation

8.12.3.1 bool Gui::Widgets::CheckFields::CheckEmail::check (const QString *text*) [virtual]

CheckEmail::check Check if the field email is valid. To be valid, an email address should be under this form: me@me.xx An email address need:

- 1 character [A-Z] or [a-z] minimum before the character </i>
- *the character '@'*
- *1 character [A-Z] or [a-z] after the character</i>minimum and before the character .*
- *1 character [A-Z] or [a-z] minimum afer the character . Return TRUE if email address is valid, else FALSE*

  Parameters

text	*Text to check*

  Returns

  boolean Validity of the text

Implements Gui::Widgets::CheckFields::ICheckField.

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

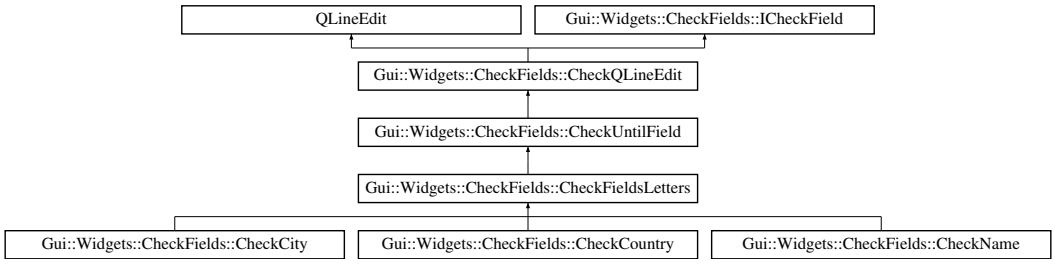
- src/gui/widgets/checkfields/checkemail.h
- src/gui/widgets/checkfields/checkemail.cpp

8.13 Gui::Widgets::CheckFields::CheckFieldsLetters Class Reference

The CheckFieldsLetters class Field with only letters (no numbers)

`#include <checkfieldsletters.h>`

Inheritance diagram for Gui::Widgets::CheckFields::CheckFieldsLetters:



Public Member Functions

- CheckFieldsLetters (QWidget ∗w=0, QPushButton ∗btn=0)

  CheckFieldsLetters::CheckFieldsLetters Construct a CheckFieldsLetters.
- bool check (QString text)

  CheckFieldsLetters::check Check if the field contains only letters.

Additional Inherited Members

8.13.1 Detailed Description

The CheckFieldsLetters class Field with only letters (no numbers)

8.13.2 Constructor & Destructor Documentation

8.13.2.1 Gui::Widgets::CheckFields::CheckFieldsLetters::CheckFieldsLetters (QWidget ∗ *w* = 0, QPushButton ∗ *btn* = 0)

CheckFieldsLetters::CheckFieldsLetters Construct a CheckFieldsLetters.

Parameters

w	QWidget linked to **CheckFieldsLetters**

8.13.3 Member Function Documentation

8.13.3.1 bool Gui::Widgets::CheckFields::CheckFieldsLetters::check (QString *text*) [virtual]

CheckFieldsLetters::check Check if the field contains only letters.

Parameters

text	Text to check

Returns

boolean Validity of the text

Implements Gui::Widgets::CheckFields::ICheckField.

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

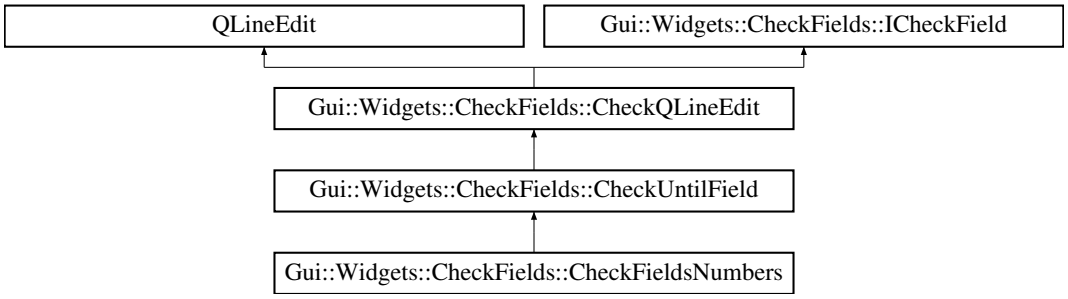
- src/gui/widgets/checkfields/checkfieldsletters.h
- src/gui/widgets/checkfields/checkfieldsletters.cpp

8.14 Gui::Widgets::CheckFields::CheckFieldsNumbers Class Reference

The CheckFieldsNumbers class Line Edit of number with a check icon.

`#include <checkfieldsnumbers.h>`

Inheritance diagram for Gui::Widgets::CheckFields::CheckFieldsNumbers:



Public Member Functions

- CheckFieldsNumbers (QWidget ∗w=0, QPushButton ∗btn=0)
  CheckFieldsNumbers::CheckFieldsNumbers Construct a CheckFieldsNumbers.
- bool check (QString text)
  CheckFieldsNumbers::check Check if the field contains only numbers.

Additional Inherited Members

8.14.1 Detailed Description

The CheckFieldsNumbers class Line Edit of number with a check icon.

Author

Florent BERBIE

See also

CheckQLineEdit
CheckUntilField

8.14.2 Constructor & Destructor Documentation

8.14.2.1 Gui::Widgets::CheckFields::CheckFieldsNumbers::CheckFieldsNumbers (QWidget ∗ *w* = 0, QPushButton ∗ *btn* = 0)

CheckFieldsNumbers::CheckFieldsNumbers Construct a CheckFieldsNumbers.

Parameters

w	QWidget linked to **CheckFieldsNumbers**

8.14.3 Member Function Documentation

8.14.3.1 bool Gui::Widgets::CheckFields::CheckFieldsNumbers::check (QString *text*) [virtual]

CheckFieldsNumbers::check Check if the field contains only numbers.

Parameters

text	Text to check

Returns

boolean Validity of the text

Implements Gui::Widgets::CheckFields::ICheckField.

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

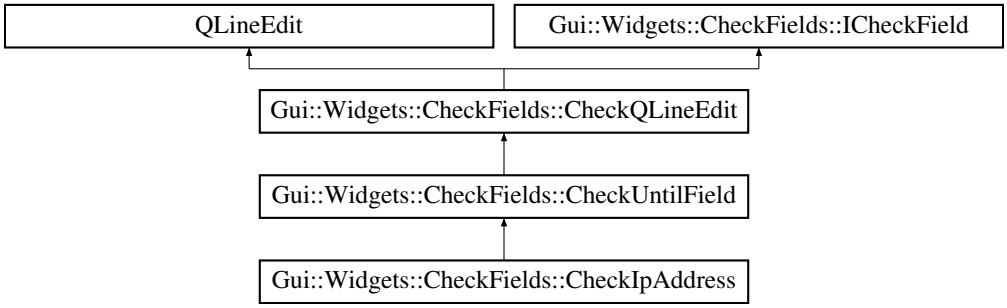
- src/gui/widgets/checkfields/checkfieldsnumbers.h
- src/gui/widgets/checkfields/checkfieldsnumbers.cpp

8.15 Gui::Widgets::CheckFields::CheckIpAddress Class Reference

The CheckIpAddress class Line Edit of IP address with a check icon.

`#include <checkipaddress.h>`

Inheritance diagram for Gui::Widgets::CheckFields::CheckIpAddress:



Public Member Functions

- CheckIpAddress (QWidget ∗w=0, QPushButton ∗btn=0)
  CheckIpAddress::CheckIpAddress Construct a CheckIpAddress.
- bool check (QString text)
  CheckIpAddress::check Check if the field contains an IP address or domain name valid.

Additional Inherited Members

8.15.1 Detailed Description

The CheckIpAddress class Line Edit of IP address with a check icon.

Author

Florent BERBIE

See also

CheckQLineEdit
CheckUntilField

8.15.2 Constructor & Destructor Documentation

8.15.2.1 Gui::Widgets::CheckFields::CheckIpAddress::CheckIpAddress (QWidget * *w* = 0, QPushButton * *btn* = 0)

CheckIpAddress::CheckIpAddress Construct a CheckIpAddress.

Parameters

w	QWidget linked to **CheckIpAddress**

8.15.3 Member Function Documentation

8.15.3.1 bool Gui::Widgets::CheckFields::CheckIpAddress::check (QString *text*) [virtual]

CheckIpAddress::check Check if the field contains an IP address or domain name valid.

Parameters

text	Text to check

Returns

boolean Validity of the text

Implements Gui::Widgets::CheckFields::ICheckField.

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

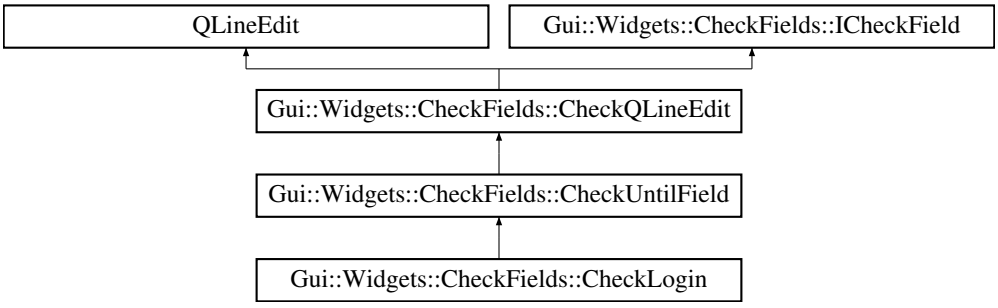
- src/gui/widgets/checkfields/checkipaddress.h
- src/gui/widgets/checkfields/checkipaddress.cpp

8.16 Gui::Widgets::CheckFields::CheckLogin Class Reference

The CheckLogin class Line Edit of login with a check icon.

`#include <checklogin.h>`

Inheritance diagram for Gui::Widgets::CheckFields::CheckLogin:



Public Slots

- void passwordPreviousInputed (const QString &text)

  CheckLogin::fieldTextChanged For each new characater inputed or removed, displays an icon to show if the field is valid or not.

Public Member Functions

- CheckLogin (QWidget ∗w=0, QPushButton ∗btn=0)

  CheckLogin::CheckLogin Construct a CheckLogin.
- bool check (QString text)

  CheckLogin::check Check if the field contains only numbers.

8.16.1 Detailed Description

The CheckLogin class Line Edit of login with a check icon.

Author

Florent BERBIE

See also

CheckQLineEdit
CheckUntilField

8.16.2 Constructor & Destructor Documentation

8.16.2.1 Gui::Widgets::CheckFields::CheckLogin::CheckLogin (QWidget ∗ *w* = 0, QPushButton ∗ *btn* = 0)

CheckLogin::CheckLogin Construct a CheckLogin.

Parameters

w	QWidget linked to **CheckIpAddress**

8.16.3 Member Function Documentation

8.16.3.1 bool Gui::Widgets::CheckFields::CheckLogin::check (QString *text*) [virtual]

CheckLogin::check Check if the field contains only numbers.

Parameters

text	Text to check

Returns

boolean Validity of the text

Implements Gui::Widgets::CheckFields::ICheckField.

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

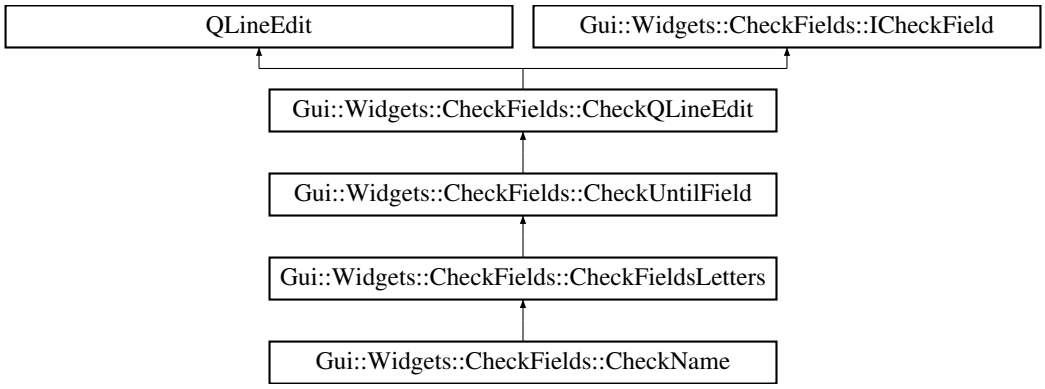
- src/gui/widgets/checkfields/checklogin.h
- src/gui/widgets/checkfields/checklogin.cpp

8.17 Gui::Widgets::CheckFields::CheckName Class Reference

The CheckName class Line edit of name with a check icon.

`#include <checkname.h>`

Inheritance diagram for Gui::Widgets::CheckFields::CheckName:



Public Member Functions

- CheckName (QWidget ∗w=0, QPushButton ∗btn=0)

  CheckName::CheckName Construct a CheckName.

Additional Inherited Members

8.17.1 Detailed Description

The CheckName class Line edit of name with a check icon.

8.17.2 Constructor & Destructor Documentation

8.17.2.1 Gui::Widgets::CheckFields::CheckName::CheckName (QWidget ∗ *w* = 0, QPushButton ∗ *btn* = 0)

CheckName::CheckName Construct a CheckName.

Parameters

w	QWidget linked to **CheckName**

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

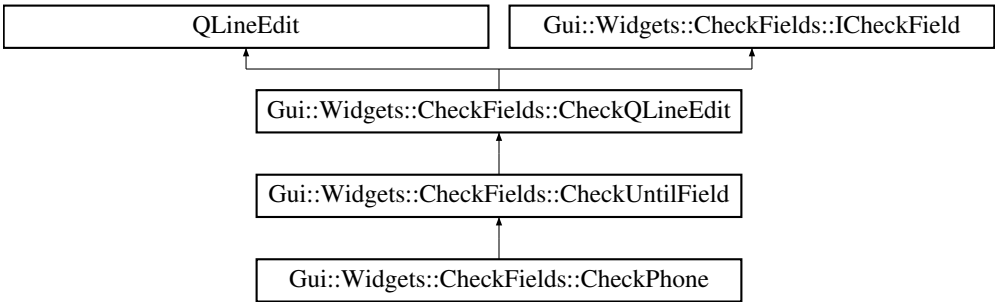
- src/gui/widgets/checkfields/checkname.h
- src/gui/widgets/checkfields/checkname.cpp

8.18 Gui::Widgets::CheckFields::CheckPhone Class Reference

The CheckPhone class Line Edit of Phone number with a check icon.

`#include <checkphone.h>`

Inheritance diagram for Gui::Widgets::CheckFields::CheckPhone:



Public Member Functions

- CheckPhone (QWidget ∗w=0, QPushButton ∗btn=0)

  CheckPhone::CheckPhone Construct a CheckPhone.
- bool check (QString text)

  CheckPhone::check Check if the field is valid. To be valid, a name should be composed of a character.
- QString getCountry () const

  CheckPhone::getCountry Return the country linked to current field.
- void setCountry (const QString &country)

  CheckPhone::setCountry Modify the country linked to field.

Additional Inherited Members

8.18.1 Detailed Description

The CheckPhone class Line Edit of Phone number with a check icon.

8.18.2 Constructor & Destructor Documentation

8.18.2.1 Gui::Widgets::CheckFields::CheckPhone::CheckPhone (QWidget ∗ *w* = 0, QPushButton ∗ *btn* = 0)

CheckPhone::CheckPhone Construct a CheckPhone.

Parameters

w	QWidget linked to **CheckPhone**

8.18.3 Member Function Documentation

8.18.3.1 bool Gui::Widgets::CheckFields::CheckPhone::check (QString *text*) [virtual]

CheckPhone::check Check if the field is valid. To be valid, a name should be composed of a character.

Parameters

text	Text to check

Returns

boolean Validity of the text

Implements Gui::Widgets::CheckFields::ICheckField.

8.18.3.2 QString Gui::Widgets::CheckFields::CheckPhone::getCountry ( ) const

CheckPhone::getCountry Return the country linked to current field.

Returns

8.18.3.3 void Gui::Widgets::CheckFields::CheckPhone::setCountry (const QString & *country*)

CheckPhone::setCountry Modify the *country* linked to field.

Parameters

country	New country

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

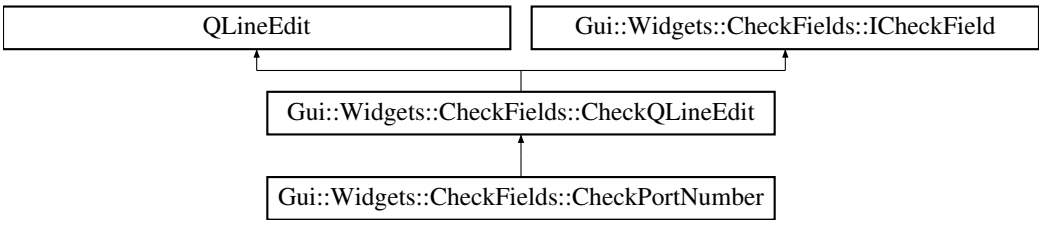
- src/gui/widgets/checkfields/checkphone.h
- src/gui/widgets/checkfields/checkphone.cpp

8.19 Gui::Widgets::CheckFields::CheckPortNumber Class Reference

The CheckFieldsNumbers class Line Edit of number with a check icon.

`#include <checkportnumber.h>`

Inheritance diagram for Gui::Widgets::CheckFields::CheckPortNumber:



Public Member Functions

- CheckPortNumber (QWidget ∗w=0, QPushButton ∗btn=0)

  CheckPortNumber.
- bool check (QString text)

  CheckPortNumber::check Check if the field contains only numbers or an empty text.

Additional Inherited Members

8.19.1 Detailed Description

The CheckFieldsNumbers class Line Edit of number with a check icon.

Author

Florent BERBIE

See also

CheckQLineEdit
CheckUntilField

8.19.2 Constructor & Destructor Documentation

8.19.2.1 Gui::Widgets::CheckFields::CheckPortNumber::CheckPortNumber (QWidget * *w* = 0, QPushButton * *btn* = 0)

CheckPortNumber.

Parameters

w	Widget parent
btn	Button parretn

8.19.3 Member Function Documentation

8.19.3.1 bool Gui::Widgets::CheckFields::CheckPortNumber::check (QString *text*) [virtual]

CheckPortNumber::check Check if the field contains only numbers or an empty text.

Parameters

text	Text to check

Returns

boolean Validity of the text

Implements Gui::Widgets::CheckFields::ICheckField.

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

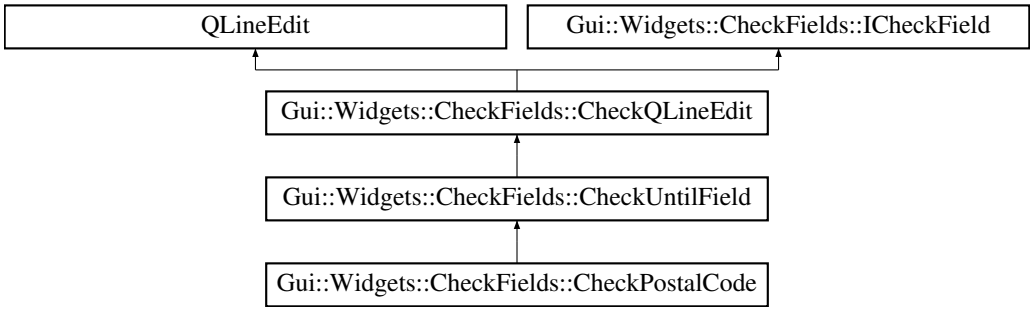
- src/gui/widgets/checkfields/checkportnumber.h
- src/gui/widgets/checkfields/checkportnumber.cpp

8.20 Gui::Widgets::CheckFields::CheckPostalCode Class Reference

The CheckPostalCode class Line Edit of postal code with a check icon.

`#include <checkpostalcode.h>`

Inheritance diagram for Gui::Widgets::CheckFields::CheckPostalCode:



Public Member Functions

- CheckPostalCode (QWidget ∗w=0, QPushButton ∗btn=0)

  CheckPostalCode::CheckPostalCode Construct a CheckPostalCode.
- bool check (QString text)

  CheckPostalCode::check Check if the field is valid. To be valid, a name should be composed of a character.
- QString getCountry () const

  CheckPostalCode::getCountry Return the country linked to current field.
- void setCountry (const QString &country)

  CheckPostalCode::setCountry Modify the country linked to field.

Additional Inherited Members

8.20.1 Detailed Description

The CheckPostalCode class Line Edit of postal code with a check icon.

8.20.2 Constructor & Destructor Documentation

8.20.2.1 Gui::Widgets::CheckFields::CheckPostalCode::CheckPostalCode (QWidget ∗ *w* = 0, QPushButton ∗ *btn* = 0)

CheckPostalCode::CheckPostalCode Construct a CheckPostalCode.

Parameters

w	QWidget linked to **CheckPostalCode**

8.20.3 Member Function Documentation

8.20.3.1 bool Gui::Widgets::CheckFields::CheckPostalCode::check (QString *text*) [virtual]

CheckPostalCode::check Check if the field is valid. To be valid, a name should be composed of a character.

Parameters

text	Text to check

Returns

boolean Validity of the text

Implements Gui::Widgets::CheckFields::ICheckField.

8.20.3.2 QString Gui::Widgets::CheckFields::CheckPostalCode::getCountry ( ) const

CheckPostalCode::getCountry Return the country linked to current field.

Returns

country Country of the field

8.20.3.3 void Gui::Widgets::CheckFields::CheckPostalCode::setCountry (const QString & *country*)

CheckPostalCode::setCountry Modify the *country* linked to field.

Parameters

country	New country

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

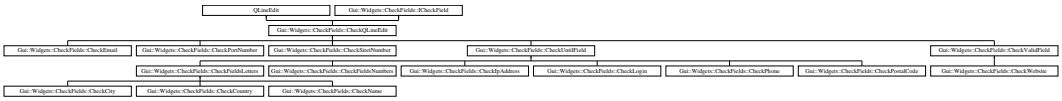
- src/gui/widgets/checkfields/checkpostalcode.h
- src/gui/widgets/checkfields/checkpostalcode.cpp

8.21 Gui::Widgets::CheckFields::CheckQLineEdit Class Reference

The CheckQLineEdit class LineEdit custom with a check of text inputed.

`#include <checkqlineedit.h>`

Inheritance diagram for Gui::Widgets::CheckFields::CheckQLineEdit:



Public Slots

- void fieldTextChanged (const QString &text)

  CheckQLineEdit::fieldTextChanged For each new characater inputed or removed, displays an icon to show if the field is valid or not.

Public Member Functions

- CheckQLineEdit (QWidget *parent=0, QPushButton *btn=0)

  CheckQLineEdit::CheckQLineEdit Construct a CheckQLineEdit.
- void displayCheckValidFieldIcon ()

  CheckQLineEdit::displayCheckValidFieldIcon Display a valid icon into the field.
- void displayCheckNoValidFieldIcon ()

  CheckQLineEdit::displayCheckNoValidFieldIcon Display a "no valid" icon into the field.
- QPushButton * getBtnValid () const

  CheckQLineEdit::getBtnValid.
- void setBtnValid (QPushButton *getBtnValid)

  CheckQLineEdit::setBtnValid.
- bool isValid ()

  isValid Return true if the current field if valid

8.21.1 Detailed Description

The CheckQLineEdit class LineEdit custom with a check of text inputed.

8.21.2 Constructor & Destructor Documentation

8.21.2.1 Gui::Widgets::CheckFields::CheckQLineEdit::CheckQLineEdit (QWidget * *parent* = `0`, QPushButton * *btn* = `0`) `[explicit]`

CheckQLineEdit::CheckQLineEdit Construct a CheckQLineEdit.

Parameters

parent	

8.21.3 Member Function Documentation

8.21.3.1 QPushButton * Gui::Widgets::CheckFields::CheckQLineEdit::getBtnValid ( ) const

CheckQLineEdit::getBtnValid.

Returns

a

8.21.3.2 bool Gui::Widgets::CheckFields::CheckQLineEdit::isValid ( )

isValid Return true if the current field if valid

Returns

boolean

8.21.3.3 void Gui::Widgets::CheckFields::CheckQLineEdit::setBtnValid (QPushButton * *getBtnValid*)

CheckQLineEdit::setBtnValid.

Parameters

getBtnValid	

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

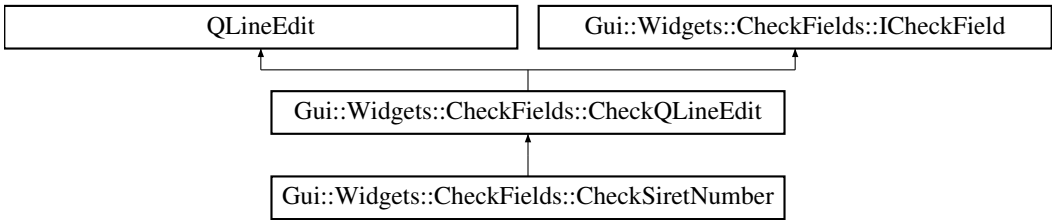
- src/gui/widgets/checkfields/checkqlineedit.h
- src/gui/widgets/checkfields/checkqlineedit.cpp

8.22 Gui::Widgets::CheckFields::CheckSiretNumber Class Reference

The CheckSiretNumber class Line Edit with a check icon.

`#include <checksiretnumber.h>`

Inheritance diagram for Gui::Widgets::CheckFields::CheckSiretNumber:



Public Member Functions

- CheckSiretNumber (QWidget *w=0, QPushButton *btn=0)

CheckSiretNumber::CheckSiretNumber *Construct a CheckSiretNumber.*

- bool check (QString text)

  CheckSiretNumber::check *Check if the field noSiret is valid. To be valid, a SIRET number should be composed of numbers.*

Additional Inherited Members

8.22.1 Detailed Description

The CheckSiretNumber class Line Edit with a check icon.

8.22.2 Constructor & Destructor Documentation

8.22.2.1 Gui::Widgets::CheckFields::CheckSiretNumber::CheckSiretNumber (QWidget * *w* = 0, QPushButton * *btn* = 0)

CheckSiretNumber::CheckSiretNumber Construct a CheckSiretNumber.

Parameters

w	QWidget linked to **CheckSiretNumber**

8.22.3 Member Function Documentation

8.22.3.1 bool Gui::Widgets::CheckFields::CheckSiretNumber::check (QString *text*) [virtual]

CheckSiretNumber::check Check if the field noSiret is valid. To be valid, a SIRET number should be composed of numbers.

Parameters

text	Text to check

Returns

boolean Validy of the text

Implements Gui::Widgets::CheckFields::ICheckField.

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

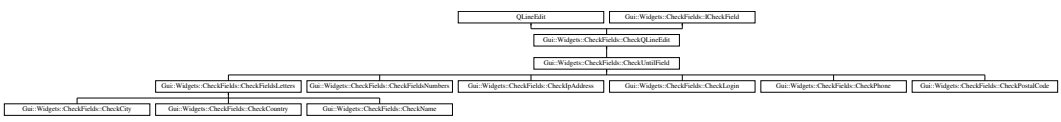
- src/gui/widgets/checkfields/checksiretnumber.h
- src/gui/widgets/checkfields/checksiretnumber.cpp

8.23 Gui::Widgets::CheckFields::CheckUntilField Class Reference

The CheckUntilField class.

`#include <checkuntilfield.h>`

Inheritance diagram for Gui::Widgets::CheckFields::CheckUntilField:



Public Member Functions

- CheckUntilField (QWidget *w=0, QPushButton *btn=0)

  CheckUntilField::CheckUntilField Construct a CheckUntilField.
- bool check (QString text)

  CheckUntilField::check Check if the field is valid. To be valid, a name should be composed of a character.

Additional Inherited Members

8.23.1 Detailed Description

The CheckUntilField class.

8.23.2 Constructor & Destructor Documentation

8.23.2.1 Gui::Widgets::CheckFields::CheckUntilField::CheckUntilField (QWidget * *w* = 0, QPushButton * *btn* = 0)

CheckUntilField::CheckUntilField Construct a CheckUntilField.

Parameters

w	QWidget linked to **CheckUntilField**

8.23.3 Member Function Documentation

8.23.3.1 bool Gui::Widgets::CheckFields::CheckUntilField::check (QString *text*) [virtual]

CheckUntilField::check Check if the field is valid. To be valid, a name should be composed of a character.

Parameters

text	Text to check

Returns

boolean Validity of the text

Implements Gui::Widgets::CheckFields::ICheckField.

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

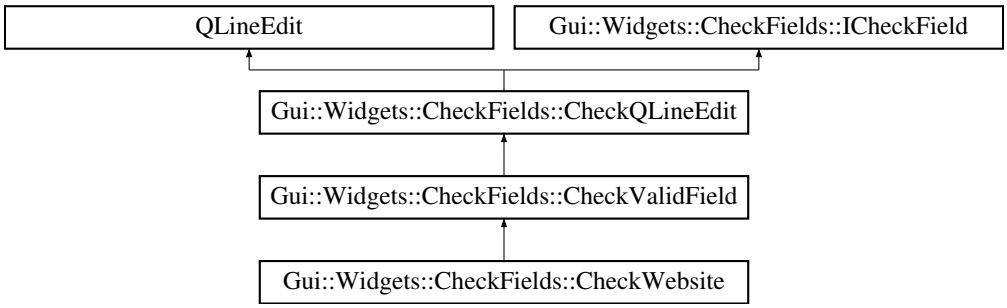
- src/gui/widgets/checkfields/checkuntilfield.h
- src/gui/widgets/checkfields/checkuntilfield.cpp

8.24 Gui::Widgets::CheckFields::CheckValidField Class Reference

The CheckValidField class Check field not required.

`#include <checkvalidfield.h>`

Inheritance diagram for Gui::Widgets::CheckFields::CheckValidField:



Public Member Functions

- CheckValidField (QWidget ∗w=0, QPushButton ∗btn=0)

  CheckValidField::CheckValidField.
- bool check (QString text)

  CheckValidField::check Return TRUE : the field is not required.

Additional Inherited Members

8.24.1 Detailed Description

The CheckValidField class Check field not required.

8.24.2 Constructor & Destructor Documentation

8.24.2.1 Gui::Widgets::CheckFields::CheckValidField::CheckValidField (QWidget ∗ *w* = 0, QPushButton ∗ *btn* = 0)

CheckValidField::CheckValidField.

Parameters

w	QWidget linked to **CheckValidField**

8.24.3 Member Function Documentation

8.24.3.1 bool Gui::Widgets::CheckFields::CheckValidField::check (QString *text*) [virtual]

CheckValidField::check Return TRUE : the field is not required.

Parameters

text	Text to check

Returns

boolean Validity of the text

Implements Gui::Widgets::CheckFields::ICheckField.

Reimplemented in Gui::Widgets::CheckFields::CheckWebsite.

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

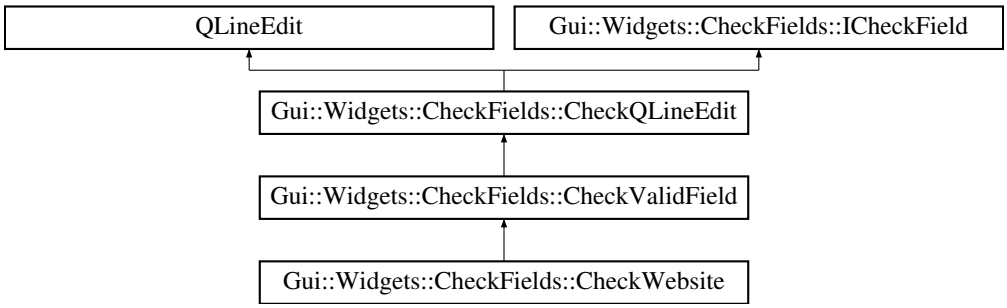
- src/gui/widgets/checkfields/checkvalidfield.h
- src/gui/widgets/checkfields/checkvalidfield.cpp

8.25 Gui::Widgets::CheckFields::CheckWebsite Class Reference

The CheckWebsite class Line Edit of website with a check icon.

`#include <checkwebsite.h>`

Inheritance diagram for Gui::Widgets::CheckFields::CheckWebsite:



Public Member Functions

- CheckWebsite (QWidget ∗w=0, QPushButton ∗btn=0)

  CheckWebsite::CheckWebsite Construct a CheckWebsite.
- bool check (QString text)

  CheckWebsite::check Check if the field contains an website address.

Additional Inherited Members

8.25.1 Detailed Description

The CheckWebsite class Line Edit of website with a check icon.

See also

CheckQLineEdit
CheckUntilField
CheckIpAddress

8.25.2 Constructor & Destructor Documentation

8.25.2.1 Gui::Widgets::CheckFields::CheckWebsite::CheckWebsite (QWidget ∗ *w* = 0, QPushButton ∗ *btn* = 0)

CheckWebsite::CheckWebsite Construct a CheckWebsite.

Parameters

w	QWidget linked to **CheckWebsite**

8.25.3 Member Function Documentation

8.25.3.1 bool Gui::Widgets::CheckFields::CheckWebsite::check (QString *text*) [virtual]

CheckWebsite::check Check if the field contains an website address.

Parameters

text	Text to check

Returns

boolean Validity of the text

Reimplemented from Gui::Widgets::CheckFields::CheckValidField.

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

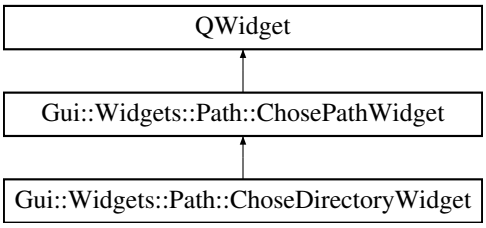
- src/gui/widgets/checkfields/checkwebsite.h
- src/gui/widgets/checkfields/checkwebsite.cpp

8.26 Gui::Widgets::Path::ChoseDirectoryWidget Class Reference

The ChoseDirectoryWidget class Open a QFileDialog.and display path in textfield.

`#include <chosedirectorywidget.h>`

Inheritance diagram for Gui::Widgets::Path::ChoseDirectoryWidget:



Public Slots

- void fillField ()

  fillField Fill the textfield with path information

Public Member Functions

- ChoseDirectoryWidget (QWidget ∗parent=0)

  ChoseDirectoryWidget Construct Widget.

Additional Inherited Members

8.26.1 Detailed Description

The ChoseDirectoryWidget class Open a QFileDialog.and display path in textfield.

8.26.2 Constructor & Destructor Documentation

8.26.2.1 Gui::Widgets::Path::ChoseDirectoryWidget::ChoseDirectoryWidget (QWidget ∗ *parent* = `0`) `[explicit]`

ChoseDirectoryWidget Construct Widget.

Parameters

parent	The parent widget

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

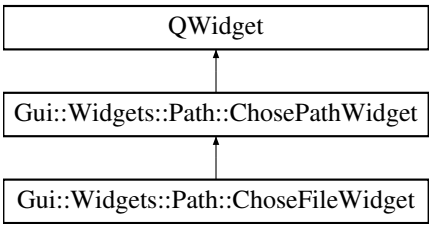
- src/gui/widgets/path/chosedirectorywidget.h
- src/gui/widgets/path/chosedirectorywidget.cpp

8.27 Gui::Widgets::Path::ChoseFileWidget Class Reference

The ChoseFileWidget class Chose a File in computer.

`#include <chosefilewidget.h>`

Inheritance diagram for Gui::Widgets::Path::ChoseFileWidget:



Public Slots

- void fillField ()

  fillField Fill the textfield with path information

Public Member Functions

- ChoseFileWidget (QWidget *parent=0)

  ChoseFileWidget Construct a choseFileWidget.
- QString getypeFiles () const

  *getypeFiles Return the types of files, by default it's **
- void setTypeFiles (const QString &getypeFiles)

  *setTypeFiles Change the files types. By default it's **

Additional Inherited Members

8.27.1 Detailed Description

The ChoseFileWidget class Chose a File in computer.

8.27.2 Constructor & Destructor Documentation

8.27.2.1 Gui::Widgets::Path::ChoseFileWidget::ChoseFileWidget (QWidget * *parent* = `0`) `[explicit]`

ChoseFileWidget Construct a choseFileWidget.

Parameters

parent	The parent widget

8.27.3 Member Function Documentation

8.27.3.1 QString Gui::Widgets::Path::ChoseFileWidget::getypeFiles ( ) const

getypeFiles Return the types of files, by default it's ∗

Returns

The type of files

8.27.3.2 void Gui::Widgets::Path::ChoseFileWidget::setTypeFiles (const QString & *getypeFiles*)

setTypeFiles Change the files types. By default it's ∗

Parameters

getypeFiles	The new files types.

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

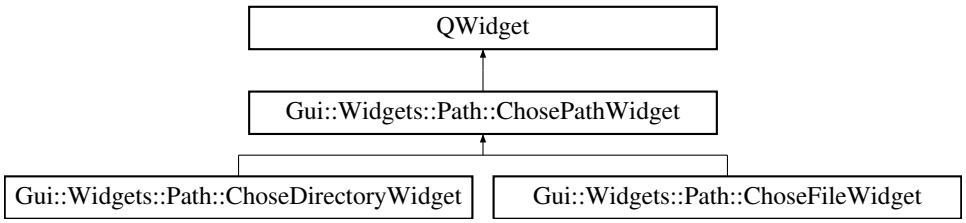
- src/gui/widgets/path/chosefilewidget.h
- src/gui/widgets/path/chosefilewidget.cpp

8.28 Gui::Widgets::Path::ChosePathWidget Class Reference

The ChoseDirectoryWidget class Open a QFileDialog.and display path in textfield.

`#include <chosepathwidget.h>`

Inheritance diagram for Gui::Widgets::Path::ChosePathWidget:



Public Slots

- virtual void fillField ()

  fillField Fill the textfield with path information

Signals

- void textChanged ()

  textChanged Signal is send when path changed.

Public Member Functions

- ChosePathWidget (QWidget *parent=0)

  ChoseFileWidget Construct a choseFileWidget.
- void setField (QString text)

  setField Change the path in textfield
- QString getField ()

  getField The text in field
- virtual QString getDefaultLocation ()

  getDefaultLocation The default location when we open popup. In default case, it's ~/Documents

Protected Attributes

- Ui::ChosePathWidget * **ui**

8.28.1 Detailed Description

The ChoseDirectoryWidget class Open a QFileDialog.and display path in textfield.

8.28.2 Constructor & Destructor Documentation

8.28.2.1 Gui::Widgets::Path::ChosePathWidget::ChosePathWidget (QWidget * *parent* = `0`) `[explicit]`

ChoseFileWidget Construct a choseFileWidget.

Parameters

parent	The parent widget

8.28.3 Member Function Documentation

8.28.3.1 QString Gui::Widgets::Path::ChosePathWidget::getDefaultLocation ( ) `[virtual]`

getDefaultLocation The default location when we open popup. In default case, it's ~/Documents

Returns

The default location

8.28.3.2 QString Gui::Widgets::Path::ChosePathWidget::getField (void)

getField The text in field

Returns

The path text

8.28.3.3 void Gui::Widgets::Path::ChosePathWidget::setField (QString *text*)

setField Change the path in textfield

Parameters

text	The new text

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

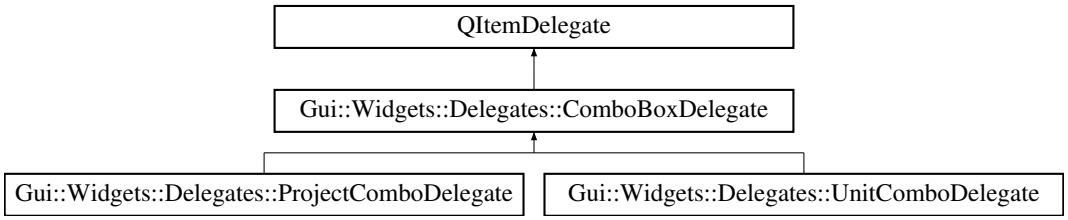
- src/gui/widgets/path/chosepathwidget.h
- src/gui/widgets/path/chosepathwidget.cpp

8.29 Gui::Widgets::Delegates::ComboBoxDelegate Class Reference

The ComboBoxDelegate class.

`#include <comboboxdelegate.h>`

Inheritance diagram for Gui::Widgets::Delegates::ComboBoxDelegate:



Public Member Functions

- ComboBoxDelegate (QObject ∗parent=0)

  ComboBoxDelegate::ComboBoxDelegate Construct a ComboBoxDelegate.
- virtual QWidget ∗ createEditor (QWidget ∗parent, const QStyleOptionViewItem &option, const QModelIndex &index) const =0

  ComboBoxDelegate::createEditor Return a ComboBox specified by index item defined by the parent widget and style option which are used to control how the editor widgets appears.
- void paint (QPainter ∗painter, const QStyleOptionViewItem &option, const QModelIndex &index) const =0

  ComboBoxDelegate::paint Renders the delegate using the given painter and style option for the item specified by index
- void setEditorData (QWidget ∗editor, const QModelIndex &index) const

  ComboBoxDelegate::setEditorData Sets the data to be displayed and edited by the editor from the data model item specified by the model index
- void setModelData (QWidget ∗editor, QAbstractItemModel ∗model, const QModelIndex &index) const

  ComboBoxDelegate::setModelData Gets data from the editor widget and stores it in the specified model at the item index
- void updateEditorGeometry (QWidget ∗editor, const QStyleOptionViewItem &option, const QModelIndex &index) const

  ComboBoxDelegate::updateEditorGeometry Update the editor for the item specified by index according to the style option given.

8.29.1 Detailed Description

The ComboBoxDelegate class.

Author

Antoine de Roquemaurel

8.29.2 Constructor & Destructor Documentation

8.29.2.1 Gui::Widgets::Delegates::ComboBoxDelegate::ComboBoxDelegate (QObject * *parent* = 0)

ComboBoxDelegate::ComboBoxDelegate Construct a ComboBoxDelegate.

Parameters

parent	Object parent

8.29.3 Member Function Documentation

8.29.3.1 virtual QWidget* Gui::Widgets::Delegates::ComboBoxDelegate::createEditor (QWidget * *parent*, const QStyleOptionViewItem & *option*, const QModelIndex & *index*) const [pure virtual]

ComboBoxDelegate::createEditor Return a ComboBox specified by *index* item defined by the *parent* widget and style *option* which are used to control how the editor widgets appears.

Parameters

parent	Widget parent
option	Option style
index	Index for editing

Returns

ComboBox

Implemented in Gui::Widgets::Delegates::UnitComboDelegate, and Gui::Widgets::Delegates::ProjectComboDelegate.

8.29.3.2 void Gui::Widgets::Delegates::ComboBoxDelegate::paint (QPainter * *painter*, const QStyleOptionViewItem & *option*, const QModelIndex & *index*) const [pure virtual]

ComboBoxDelegate::paint Renders the delegate using the given *painter* and style *option* for the item specified by *index*

Parameters

parent	Widget parent
option	Option style
index	Index for editing

Implemented in Gui::Widgets::Delegates::UnitComboDelegate, and Gui::Widgets::Delegates::ProjectComboDelegate.

8.29.3.3 void Gui::Widgets::Delegates::ComboBoxDelegate::setEditorData (QWidget * *editor*, const QModelIndex & *index*) const

ComboBoxDelegate::setEditorData Sets the data to be displayed and edited by the *editor* from the data model item specified by the model *index*

Parameters

editor	Data edited

index	Index of the model to edit

8.29.3.4 void Gui::Widgets::Delegates::ComboBoxDelegate::setModelData (QWidget * *editor*, QAbstractItemModel * *model*, const QModelIndex & *index*) const

ComboBoxDelegate::setModelData Gets data from the *editor* widget and stores it in the specified *model* at the item *index*

Parameters

editor	Editor Widget
model	Model to store data
index	Item index

8.29.3.5 void Gui::Widgets::Delegates::ComboBoxDelegate::updateEditorGeometry (QWidget * *editor*, const QStyleOptionViewItem & *option*, const QModelIndex & *index*) const

ComboBoxDelegate::updateEditorGeometry Update the *editor* for the item specified by *index* according to the style *option* given.

Parameters

editor	Editor widget to update
option	Style option
index	Item index

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

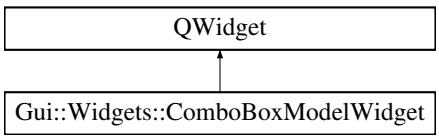
- src/gui/widgets/delegates/comboboxdelegate.h
- src/gui/widgets/delegates/comboboxdelegate.cpp

8.30 Gui::Widgets::ComboBoxModelWidget Class Reference

The ComboBoxModelWidget class Model of ComboBox.

`#include <comboboxmodelwidget.h>`

Inheritance diagram for Gui::Widgets::ComboBoxModelWidget:



Public Member Functions

- ComboBoxModelWidget (QWidget *parent=0)

  ComboBoxModelWidget::ComboBoxModelWidget Construct a ComboBoxModelWidget.

8.30.1 Detailed Description

The ComboBoxModelWidget class Model of ComboBox.

8.30.2 Constructor & Destructor Documentation

8.30.2.1 Gui::Widgets::ComboBoxModelWidget::ComboBoxModelWidget (QWidget * *parent* = 0) [explicit]

ComboBoxModelWidget::ComboBoxModelWidget Construct a ComboBoxModelWidget.

Parameters

parent	QWidget parent

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

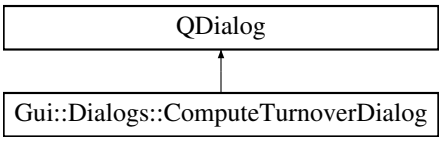
- src/gui/widgets/comboboxmodelwidget.h
- src/gui/widgets/comboboxmodelwidget.cpp

8.31 Gui::Dialogs::ComputeTurnoverDialog Class Reference

The ComputeTurnoverDialog class window to compute a turnover with a period.

`#include <computeturnoverdialog.h>`

Inheritance diagram for Gui::Dialogs::ComputeTurnoverDialog:



Public Slots

- void computeTurnover ()

  ComputeTurnoverDialog::computeTurnover compute the turnover between chosen dates in the window.
- void endDateControl (const QDate end)

  ComputeTurnoverDialog::endDateControl controls if the end date field is valid.
- void beginDateControl (const QDate begin)

  ComputeTurnoverDialog::beginDateControl controls if the begin date field is valid.

Public Member Functions

- **ComputeTurnoverDialog** (QWidget *parent=0)
- void fillLabels (const int nbBillings, const int turnover)

  ComputeTurnoverDialog::fillLabels Fills the labels with nbBillings and turnover

8.31.1 Detailed Description

The ComputeTurnoverDialog class window to compute a turnover with a period.

Author

Manantsoa Razanajatovo

8.31.2 Member Function Documentation

8.31.2.1 void Gui::Dialogs::ComputeTurnoverDialog::beginDateControl (const QDate *begin*) [slot]

ComputeTurnoverDialog::beginDateControl controls if the *begin* date field is valid.

Parameters

begin	

8.31.2.2 void Gui::Dialogs::ComputeTurnoverDialog::endDateControl (const QDate *end*) [slot]

ComputeTurnoverDialog::endDateControl controls if the *end* date field is valid.

Parameters

end	

8.31.2.3 void Gui::Dialogs::ComputeTurnoverDialog::fillLabels (const int *nbBillings*, const int *turnover*)

ComputeTurnoverDialog::fillLabels Fills the labels with *nbBillings* and *turnover*

Parameters

nbBillings	the number of Billings
turnover	the turnover computed

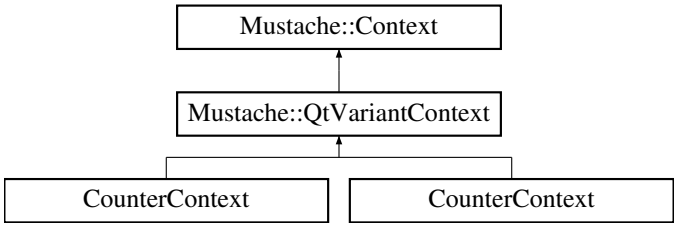
The documentation for this class was generated from the following files:

- src/gui/dialogs/computeturnoverdialog.h
- src/gui/dialogs/computeturnoverdialog.cpp

8.32 Mustache::Context Class Reference

`#include <mustache.h>`

Inheritance diagram for Mustache::Context:



Public Member Functions

- Context (PartialResolver ∗resolver=0)
- virtual QString stringValue (const QString &key) const =0
- virtual bool isFalse (const QString &key) const =0
- virtual int listCount (const QString &key) const =0
- virtual void push (const QString &key, int index=-1)=0
- virtual void pop ()=0
- QString partialValue (const QString &key) const
- PartialResolver ∗ partialResolver () const
- virtual bool canEval (const QString &key) const
- virtual QString eval (const QString &key, const QString &_template, Renderer ∗renderer)

8.32.1 Detailed Description

Context is an interface that Mustache::Renderer::render() uses to fetch substitutions for template tags.

8.32.2 Constructor & Destructor Documentation

8.32.2.1 Context::Context (PartialResolver * *resolver* = 0) [explicit]

Create a context. `resolver` is used to fetch the expansions for any {{>partial}} tags which appear in a template.

8.32.3 Member Function Documentation

8.32.3.1 bool Context::canEval (const QString & *key*) const [virtual]

Returns true if eval() should be used to render section tags using `key`. If canEval() returns true for a key, the renderer will pass the literal, unrendered block of text for the section to eval() and replace the section with the result.

canEval() and eval() are equivalents for callable objects (eg. lambdas) in other Mustache implementations.

The default implementation always returns false.

Reimplemented in CounterContext, CounterContext, and Mustache::QtVariantContext.

8.32.3.2 QString Context::eval (const QString & *key*, const QString & *_template*, Renderer * *renderer*) [virtual]

Callback used to render a template section with the given `key`. `renderer` will substitute the original section tag with the result of eval().

The default implementation returns an empty string.

Reimplemented in CounterContext, CounterContext, and Mustache::QtVariantContext.

8.32.3.3 virtual bool Mustache::Context::isFalse (const QString & *key*) const [pure virtual]

Returns true if the value for `key` is 'false' or an empty list. 'False' values typically include empty strings, the boolean value false etc.

When processing a section Mustache tag, the section is not rendered if the key is false, or for an inverted section tag, the section is only rendered if the key is false.

Implemented in Mustache::QtVariantContext.

8.32.3.4 virtual int Mustache::Context::listCount (const QString & *key*) const [pure virtual]

Returns the number of items in the list value for `key` or 0 if the value for `key` is not a list.

Implemented in Mustache::QtVariantContext.

8.32.3.5 PartialResolver * Context::partialResolver ( ) const

Returns the partial resolver passed to the constructor.

8.32.3.6 QString Context::partialValue (const QString & *key*) const

Returns the partial template for a given `key`.

8.32.3.7 virtual void Mustache::Context::pop ( ) [pure virtual]

Exit the current context.

Implemented in Mustache::QtVariantContext.

8.32.3.8 virtual void Mustache::Context::push (const QString & *key*, int *index* = -1) [pure virtual]

Set the current context to the value for `key`. If index is >= 0, set the current context to the `index'th` value in the list value for `key`.

Implemented in Mustache::QtVariantContext.

8.32.3.9 virtual QString Mustache::Context::stringValue (const QString & *key*) const [pure virtual]

Returns a string representation of the value for `key` in the current context. This is used to replace a Mustache value tag.

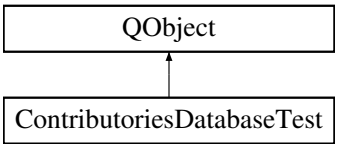
Implemented in CounterContext, CounterContext, and Mustache::QtVariantContext.

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

- src/libs/qt-mustache/src/mustache.h
- src/libs/qt-mustache/src/mustache.cpp

8.33 ContributoriesDatabaseTest Class Reference

Inheritance diagram for ContributoriesDatabaseTest:



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

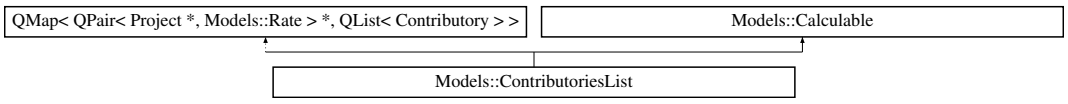
- tests/database/contributoriesdatabasetest.h
- tests/database/contributoriesdatabasetest.cpp

8.34 Models::ContributoriesList Class Reference

The ContributoriesList class List of contributories.

`#include <contributorieslist.h>`

Inheritance diagram for Models::ContributoriesList:



Public Member Functions

- ContributoriesList ()

  ContributoriesList::ContributoriesList Construct a ContributoriesList.
- double getPrice (bool isPaied=false)

  getPrice Return the price of a contributories list
- double getPrice (Models::Project *project)

  getPrice Return price of project
- double getSumQuantity ()

  ContributoriesList::getSumQuantity Return the sum of quantity (number of hours) of the Contributories.
- double getSumQuantity (Models::Project *project)

  ContributoriesList::getSumQuantity Return the sum of quantity (number of hours) of the Contributories of project.
- Models::Rate getRate (Models::Project *project)

  ContributoriesList::getRate.
- virtual void commit ()

  ContributoriesList::commit Update or insert data into the database.
- void addContributory (Models::Contributory &contributory)

  ContributoriesList::addContributory Add a new contributory
- void addProject (Project *p, Models::Rate rate)

  ContributoriesList::addProject Add a Project p and it rate
- QList< Contributory > & getContributories (Project *p)

  ContributoriesList::getContributories Return a list of Contributories for the Project p
- int getIdBilling () const

  ContributoriesList::getIdBilling Return the Billing ID.
- void setIdBilling (int idBilling)

  ContributoriesList::setIdBilling Change the Billing id by the new idBilling
- void setAllIdContributories (int idContributory)

  ContributoriesList::setAllIdContributories Change all Contributory id with the same id.
- bool isInsert () const

  ContributoriesList::isInsert Return TRUE if an element is inserting else FALSE.
- void setInsert (bool insert)

  ContributoriesList::setInsert Change the state of insertion.
- int getNbProjects ()

  ContributoriesList::getNbProjects Return the number of projects.
- QSharedPointer< Customer > getCustomer ()

  ContributoriesList::getCustomer Return the Customers linked to theses contributories.
- QList< Project * > getProjects ()

  ContributoriesList::getProjects List of Projects.
- QList< Contributory > * getAllContributories ()

  ContributoriesList::getAllContributories List of all contributories (all contributories from all projects)
- QVariantList getDataMap ()

  ContributoriesList::getDataMap Return a list of Billing and it value linked which indicates if it is inserting or not.

8.34.1 Detailed Description

The ContributoriesList class List of contributories.

Author

Antoine de Roquemaurel

8.34.2 Member Function Documentation

8.34.2.1 void Models::ContributoriesList::addContributory (Models::Contributory & *contributory*)

ContributoriesList::addContributory Add a new *contributory*

Parameters

contributory	Contributory to add

8.34.2.2 void Models::ContributoriesList::addProject (Project * *p*, Models::Rate *rate*)

ContributoriesList::addProject Add a Project *p* and it *rate*

Parameters

p	Project to add
rate	Rate of the project

8.34.2.3 QList< Contributory > * Models::ContributoriesList::getAllContributories ( )

ContributoriesList::getAllContributories List of all contributories (all contributories from all projects)

Returns

List of all contributories

8.34.2.4 QList< Contributory > & Models::ContributoriesList::getContributories (Project * *p*)

ContributoriesList::getContributories Return a list of Contributories for the Project *p*

Parameters

p	Project

Returns

List of Contributories for a project

8.34.2.5 QSharedPointer< Customer > Models::ContributoriesList::getCustomer ( )

ContributoriesList::getCustomer Return the Customers linked to theses contributories.

Returns

Customer

8.34.2.6 QVariantList Models::ContributoriesList::getDataMap ( )

ContributoriesList::getDataMap Return a list of Billing and it value linked which indicates if it is inserting or not.

Returns

List of billing and value linked

8.34.2.7 int Models::ContributoriesList::getIdBilling ( ) const

ContributoriesList::getIdBilling Return the Billing ID.

Returns

Billing id

8.34.2.8 int Models::ContributoriesList::getNbProjects ( )

ContributoriesList::getNbProjects Return the number of projects.

Returns

Count number of project

8.34.2.9 double Models::ContributoriesList::getPrice (bool *isPaied* = `false`) [virtual]

getPrice Return the price of a contributories list

Returns

The price

Implements Models::Calculable.

8.34.2.10 double Models::ContributoriesList::getPrice (Models::Project * *project*)

getPrice Return price of project

Parameters

project	The project

Returns

The price

8.34.2.11 QList< Project * > Models::ContributoriesList::getProjects (void)

ContributoriesList::getProjects List of Projects.

Returns

List of Projects

8.34.2.12 Models::Rate Models::ContributoriesList::getRate (Models::Project * *project*)

ContributoriesList::getRate.

Parameters

project	

Returns

8.34.2.13 double Models::ContributoriesList::getSumQuantity ( ) [virtual]

ContributoriesList::getSumQuantity Return the sum of quantity (number of hours) of the Contributories.

Returns

sum of quantity in days

Implements Models::Calculable.

8.34.2.14 double Models::ContributoriesList::getSumQuantity (Models::Project * *project*)

ContributoriesList::getSumQuantity Return the sum of quantity (number of hours) of the Contributories of project.

Parameters

project	The project

Returns

sum of quantity in days

8.34.2.15 bool Models::ContributoriesList::isInsert ( ) const

ContributoriesList::isInsert Return TRUE if an element is inserting else FALSE.

Returns

boolean

8.34.2.16 void Models::ContributoriesList::setAllIdContributories (int *idContributory*)

ContributoriesList::setAllIdContributories Change all Contributory id with the same id.

Parameters

idContributory	the new Contributory id

8.34.2.17 void Models::ContributoriesList::setIdBilling (int *idBilling*)

ContributoriesList::setIdBilling Change the Billing id by the new *idBilling*

Parameters

idBilling	Billind id

8.34.2.18 void Models::ContributoriesList::setInsert (bool *insert*)

ContributoriesList::setInsert Change the state of insertion.

Parameters

insert	Boolean

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

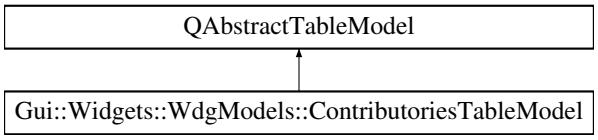
- src/models/contributorieslist.h
- src/models/contributorieslist.cpp

8.35 Gui::Widgets::WdgModels::ContributoriesTableModel Class Reference

The ContributoriesTableModel class for a custom table for contributories widget.

`#include <contributoriestablemodel.h>`

Inheritance diagram for Gui::Widgets::WdgModels::ContributoriesTableModel:



Public Member Functions

- ContributoriesTableModel (QObject *parent=0)

  ContributoriesTableModel::ContributoriesTableModel Construct a ContributoriesTableModel.
- int rowCount (const QModelIndex &) const

  ContributoriesTableModel::rowCount Number of contributories row.
- int columnCount (const QModelIndex &) const

  ContributoriesTableModel::columnCount Number of column of a contributory.
- QVariant data (const QModelIndex &index, int role) const

  ContributoriesTableModel::data Obtains data of a specify cell.
- QVariant headerData (int section, Qt::Orientation orientation, int role) const

  ContributoriesTableModel::headerData Obtains header title of table.
- bool setData (const QModelIndex &index, const QVariant &value, int role=Qt::EditRole)

  ContributoriesTableModel::setData Change data of a cell.
- void append (const Contributory &contributory)

  ContributoriesTableModel::append Add a new line in table.
- void remove (const int i)

  ContributoriesTableModel::remove Remove a line.
- Qt::ItemFlags flags (const QModelIndex &index) const

  ContributoriesTableModel::flags Differents table flags.
- QList< Contributory > getContributories ()

  ContributoriesTableModel::getContributories Get all contributories of table.
- int count ()

  ContributoriesTableModel::count Number of contributories in table.
- double getSumQuantity () const

  ContributoriesTableModel::getSumQuantity Return the sum of the hours of all contriburoies added.
- void clear ()

  ContributoriesTableModel::clear Remove all contributories.

8.35.1 Detailed Description

The ContributoriesTableModel class for a custom table for contributories widget.

Author

Antoine de Roquemaurel

See also

Contributory

8.35.2 Constructor & Destructor Documentation

8.35.2.1 Gui::Widgets::WdgModels::ContributoriesTableModel::ContributoriesTableModel (QObject * *parent* = 0)

ContributoriesTableModel::ContributoriesTableModel Construct a ContributoriesTableModel.

Parameters

parent	Parent widget

8.35.3 Member Function Documentation

8.35.3.1 void Gui::Widgets::WdgModels::ContributoriesTableModel::append (const Contributory & *contributory*)

ContributoriesTableModel::append Add a new line in table.

Parameters

contributory	The new contributory

8.35.3.2 int Gui::Widgets::WdgModels::ContributoriesTableModel::columnCount (const QModelIndex &) const

ContributoriesTableModel::columnCount Number of column of a contributory.

Returns

The number of column

8.35.3.3 int Gui::Widgets::WdgModels::ContributoriesTableModel::count ( )

ContributoriesTableModel::count Number of contributories in table.

Returns

The number of contributories

8.35.3.4 QVariant Gui::Widgets::WdgModels::ContributoriesTableModel::data (const QModelIndex & *index*, int *role*) const

ContributoriesTableModel::data Obtains data of a specify cell.

Parameters

index	The cell who we want data
role	The role of set

Returns

The data of cell

8.35.3.5 Qt::ItemFlags Gui::Widgets::WdgModels::ContributoriesTableModel::flags (const QModelIndex & *index*) const

ContributoriesTableModel::flags Differents table flags.

Parameters

index	The cell who we want to know flags

Returns

Flags

8.35.3.6 QList< Contributory > Gui::Widgets::WdgModels::ContributoriesTableModel::getContributories ( )

ContributoriesTableModel::getContributories Get all contributories of table.

Returns

The contributory list

8.35.3.7 double Gui::Widgets::WdgModels::ContributoriesTableModel::getSumQuantity ( ) const

ContributoriesTableModel::getSumQuantity Return the sum of the hours of all contriburoies added.

Returns

sum quantity

8.35.3.8 QVariant Gui::Widgets::WdgModels::ContributoriesTableModel::headerData (int *section*, Qt::Orientation *orientation*, int *role*) const

ContributoriesTableModel::headerData Obtains header title of table.

Parameters

section	The number of column
orientation	The table orientation
role	

Returns

The Title header of column

8.35.3.9 void Gui::Widgets::WdgModels::ContributoriesTableModel::remove (const int *i*)

ContributoriesTableModel::remove Remove a line.

Parameters

i	The number of line to remove

8.35.3.10 int Gui::Widgets::WdgModels::ContributoriesTableModel::rowCount (const QModelIndex &) const

ContributoriesTableModel::rowCount Number of contributories row.

Returns

The number of contributories

8.35.3.11 bool Gui::Widgets::WdgModels::ContributoriesTableModel::setData (const QModelIndex & *index*, const QVariant & *value*, int *role* = `Qt::EditRole`)

ContributoriesTableModel::setData Change data of a cell.

Parameters

index	The cell to change data
value	The new value
role	THe role of cell

Returns

True if we could edit

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

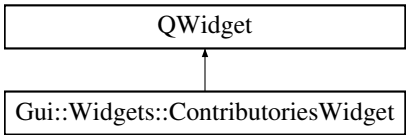
- src/gui/widgets/widgetsmodels/contributoriestablemodel.h
- src/gui/widgets/widgetsmodels/contributoriestablemodel.cpp

8.36 Gui::Widgets::ContributoriesWidget Class Reference

The ContributoriesWidget class Widget of Contributories.

`#include <contributorieswidget.h>`

Inheritance diagram for Gui::Widgets::ContributoriesWidget:



Public Slots

- void add (void)

  ContributoriesWidget::add Add a new empty contributory.
- void remove (void)

  ContributoriesWidget::remove Remove the current contributory.
- void addProject (QPair< Project *, Rate > *p=0)

  ContributoriesWidget::addProject Add a Projet and it rate p
- void removeProject (void)

  ContributoriesWidget::removeProject Remove the current Project.
- void changeProject (void)

  ContributoriesWidget::changeProject Change the current Project.
- void editing (void)

  ContributoriesWidget::editing Remove the current Project in the combobox not used.
- void updateUi (void)

  ContributoriesWidget::updateUi Update the User Interface.
- void updatePrice (void)

  ContributoriesWidget::updatePrice Update total price.

Signals

- void contributoryChanged ()

  ContributoriesWidget::contributoryChanged Signal that a contributory has changed.

Public Member Functions

- ContributoriesWidget (QSharedPointer< Customer > c, QWidget *parent=0)

  ContributoriesWidget::ContributoriesWidget Construct a ContributoriesWidget.
- ContributoriesList * getContributories () const

  ContributoriesWidget::getContributories Get contributories List.
- int count ()

  ContributoriesWidget::count Numbers of contributories.
- void add (ContributoriesList &list)

  ContributoriesWidget::add Add contributorieslist list in the model.

8.36.1 Detailed Description

The ContributoriesWidget class Widget of Contributories.

8.36.2 Constructor & Destructor Documentation

8.36.2.1 Gui::Widgets::ContributoriesWidget::ContributoriesWidget (QSharedPointer< Customer > *c*, QWidget * *parent* = 0) [explicit]

ContributoriesWidget::ContributoriesWidget Construct a ContributoriesWidget.

Parameters

c	Customer
parent	Widget parent

8.36.3 Member Function Documentation

8.36.3.1 void Gui::Widgets::ContributoriesWidget::add (ContributoriesList & *list*)

ContributoriesWidget::add Add contributorieslist *list* in the model.

Parameters

list	the **ContributoriesList**

8.36.3.2 void Gui::Widgets::ContributoriesWidget::addProject (QPair< Project *, Rate > * *p* = 0) [slot]

ContributoriesWidget::addProject Add a Projet and it rate *p*

Parameters

p	Rate linked to Project

8.36.3.3 int Gui::Widgets::ContributoriesWidget::count ( )

ContributoriesWidget::count Numbers of contributories.

Returns

Numbers of contributories

8.36.3.4 ContributoriesList * Gui::Widgets::ContributoriesWidget::getContributories ( ) const

ContributoriesWidget::getContributories Get contributories List.

Returns

ContributoriesList

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

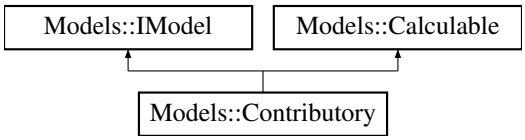
- src/gui/widgets/contributorieswidget.h
- src/gui/widgets/contributorieswidget.cpp

8.37 Models::Contributory Class Reference

The Unit enum Unity of work : hour or day.

`#include <contributory.h>`

Inheritance diagram for Models::Contributory:



Public Member Functions

- Contributory ()

  Contributory::Contributory Contruct a Contributory.
- Contributory (int id)

  Contributory::Contributory Contruct a Contributory and get data in database.
- ~Contributory ()

  Destroy an contributory object.
- void commit ()

  Contributory::commit Update or insert a contributory to the database.
- void hydrat (int id)

  Contributory::hydrat Get data about the Contributory which is specified by the identify id
- void remove ()

  Contributory::remove Remove the current Contributory.
- double getPrice (const bool paied=false)

  getPrice Return the price of a contributory
- double getSumQuantity ()

  ContributoriesList::getSumQuantity Return the sum of quantity (number of hours) of the Contributories.
- QVariantHash getDataMap ()

  getDataMap Get all data of model with a HashMap key/value
- Project * getProject () const

  Contributory::getProject Return the project linked to this Contributory.
- void setProject (Project *id)

  Contributory::setProject Modify the identify id of the Project linked to this Contributory.
- double getQuantity () const

getNbHours Number of work hour of a contributory

- void setQuantity (double value)

  setNbHours Change nbHours
- QString getDescription () const

  getDescription Description of a contributory
- void setDescription (const QString &getDescription)

  setDescription Change the contributory description
- bool operator== (const Contributory &c)

  *operator == define the operator "==" to compare two **Contributory***
- bool operator!= (const Contributory &c)

  *operator != define the operator "!=" to compare two **Contributory***
- QString getLongDescription () const

  getLongDescription A contributory has a long description : display in tex appendix
- void setLongDescription (const QString &getLongDescription)

  setLongDescription Change the long description
- Unit getUnit () const

  getUnit Return the unit (hour or day) of contributory
- void setUnit (const Unit &value)

  setUnit Change the unit
- double getHourlyRate () const

  getHourlyRate Hourly rate for this contributory
- void setHourlyRate (double value)

  setHourlyRate Change the hourly rate for this contributory

Additional Inherited Members

8.37.1 Detailed Description

The Unit enum Unity of work : hour or day.

Author

The Contributory class

8.37.2 Constructor & Destructor Documentation

8.37.2.1 Models::Contributory::Contributory (int *id*)

Contributory::Contributory Contruct a Contributory and get data in database.

Parameters

id	Contributory's id

8.37.3 Member Function Documentation

8.37.3.1 QVariantHash Models::Contributory::getDataMap ( ) [virtual]

getDataMap Get all data of model with a HashMap key/value

Returns

Model's data

Implements Models::IModel.

8.37.3.2 QString Models::Contributory::getDescription ( ) const

getDescription Description of a contributory

Returns

The description

8.37.3.3 double Models::Contributory::getHourlyRate ( ) const

getHourlyRate Hourly rate for this contributory

Returns

The hourly rate

8.37.3.4 QString Models::Contributory::getLongDescription ( ) const

getLongDescription A contributory has a long description : display in tex appendix

Returns

The long description

8.37.3.5 double Models::Contributory::getPrice (const bool *paied* = `false`) [virtual]

getPrice Return the price of a contributory

Returns

The price

Implements Models::Calculable.

8.37.3.6 Project * Models::Contributory::getProject ( ) const

Contributory::getProject Return the project linked to this Contributory.

Returns

Project linked to this Contributory

8.37.3.7 double Models::Contributory::getQuantity ( ) const

getNbHours Number of work hour of a contributory

Returns

Then number of hours

8.37.3.8 double Models::Contributory::getSumQuantity ( ) [virtual]

ContributoriesList::getSumQuantity Return the sum of quantity (number of hours) of the Contributories.

Returns

sum of quantity in hours

Implements Models::Calculable.

8.37.3.9 Unit Models::Contributory::getUnit ( ) const

getUnit Return the unit (hour or day) of contributory

Returns

The unit

8.37.3.10 void Models::Contributory::hydrat (int *id*) [virtual]

Contributory::hydrat Get data about the Contributory which is specified by the identify *id*

Parameters

id	Contributory identify

Implements Models::IModel.

8.37.3.11 bool Models::Contributory::operator!= (const Contributory & *c*)

operator != define the operator "!=" to compare two **Contributory**

Parameters

c	the **Contributory** to compare with the current **Contributory**

Returns

true if the **Contributory** are different else false

8.37.3.12 bool Models::Contributory::operator== (const Contributory & *c*)

operator == define the operator "==" to compare two **Contributory**

Parameters

c	the **Contributory** to compare with the current **Contributory**

Returns

true if the **Contributory** are equals else false

8.37.3.13 void Models::Contributory::setDescription (const QString & *getDescription*)

setDescription Change the contributory description

Parameters

getDescription	The new description

8.37.3.14 void Models::Contributory::setHourlyRate (double *value*)

setHourlyRate Change the hourly rate for this contributory

Parameters

value	The hourly rate

8.37.3.15 void Models::Contributory::setLongDescription (const QString & *getLongDescription*)

setLongDescription Change the long description

Parameters

getLong↩ Description	The new description

8.37.3.16 void Models::Contributory::setProject (Project * *id*)

Contributory::setProject Modify the identify *id* of the Project linked to this Contributory.

Parameters

id	Project Identify

8.37.3.17 void Models::Contributory::setQuantity (double *value*)

setNbHours Change nbHours

Parameters

value	The new value of nbHours

8.37.3.18 void Models::Contributory::setUnit (const Unit & *value*)

setUnit Change the unit

Parameters

value	The new unit

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

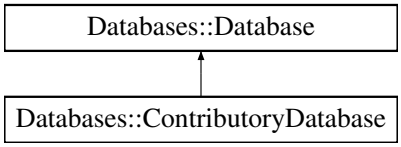
- src/models/contributory.h
- src/models/contributory.cpp

8.38 Databases::ContributoryDatabase Class Reference

The **ContributoryDatabase** class Contributory (or Quote) table database.

`#include <contributorydatabase.h>`

Inheritance diagram for Databases::ContributoryDatabase:



Public Member Functions

- Models::Contributory * getContributory (const int idContributory)
  ContributoryDatabase::getCustomer get informations about the Contributory identified by pId
- Models::ContributoriesList getContributoriesByBilling (const int billingId)
  ContributoryDatabase::getContributoriesByBilling get informations about the Contributory identified by Billing
- int addContributory (const Models::Contributory &)
  ContributoryDatabase::addContributory Add the Contributory pContributory to the database.
- void updateContributory (const Models::Contributory &)
  ContributoryDatabase::updateCustomer Update informations about the Contributory pCustomer
- void removeContributory (const int pId)
  ContributoryDatabase::removeCustomer Remove the Contributory with the id pId
- Models::Contributory * getContributory (QSqlQuery &q)
  getContributory Obtain a contributory without new query
- Models::ContributoriesList getContributoriesByBillingAndProject (const int billingId, const int projectId)
  getContributory Get contributories list by project and billing

Static Public Member Functions

- static ContributoryDatabase * instance () throw (DbException*)
  ContributoryDatabase::getInstance Return an instance of ***ContributoryDatabase***

Additional Inherited Members

8.38.1 Detailed Description

The **ContributoryDatabase** class Contributory (or Quote) table database.

Author

See also

Database
Contributory/Quote

8.38.2 Member Function Documentation

8.38.2.1 int Databases::ContributoryDatabase::addContributory (const Models::Contributory & *pContributory*)

ContributoryDatabase::addContributory Add the Contributory *pContributory* to the database.

Returns

Contributory id

8.38.2.2 Models::ContributoriesList Databases::ContributoryDatabase::getContributoriesByBilling (const int *billingId*)

ContributoryDatabase::getContributoriesByBilling get informations about the Contributory identified by *Billing*

Parameters

idBilling	Contributory id

Returns

the Contributory

8.38.2.3 Models::ContributoriesList Databases::ContributoryDatabase::getContributoriesByBillingAndProject (const int *billingId*, const int *projectId*)

getContributory Get contributories list by project and billing

Parameters

billingId	
projectId	

Returns

The contributories list by project and billing

8.38.2.4 Models::Contributory * Databases::ContributoryDatabase::getContributory (const int *idContributory*)

ContributoryDatabase::getCustomer get informations about the Contributory identified by *pId*

Parameters

idContributory	Contributory id

Returns

the Contributory

8.38.2.5 Models::Contributory * Databases::ContributoryDatabase::getContributory (QSqlQuery & *q*)

getContributory Obtain a contributory without new query

Parameters

q	The query to use

Returns

The contributory linked to q

8.38.2.6 ContributoryDatabase * Databases::ContributoryDatabase::instance ( ) throw DbException *) [static]

ContributoryDatabase::getInstance Return an instance of **ContributoryDatabase**

See also

DbException

Returns

Instance of ContributoryDatabase

8.38.2.7 void Databases::ContributoryDatabase::removeContributory (const int *pId*)

ContributoryDatabase::removeCustomer Remove the Contributory with the id *pId*

Parameters

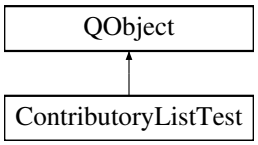
pId	Contributory id

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

- src/database/contributorydatabase.h
- src/database/contributorydatabase.cpp

8.39 ContributoryListTest Class Reference

Inheritance diagram for ContributoryListTest:

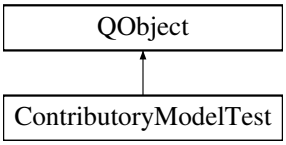


The documentation for this class was generated from the following files:

- tests/models/contributorylisttest.h
- tests/models/contributorylisttest.cpp

8.40 ContributoryModelTest Class Reference

Inheritance diagram for ContributoryModelTest:

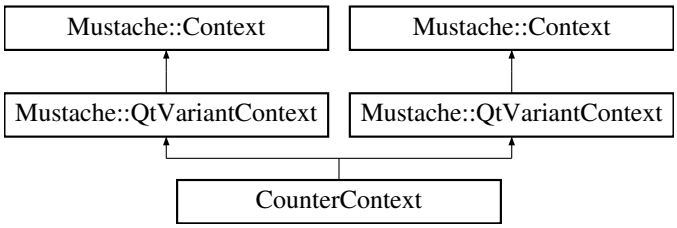


The documentation for this class was generated from the following files:

- tests/models/contributorymodeltest.h
- tests/models/contributorymodeltest.cpp

8.41 CounterContext Class Reference

Inheritance diagram for CounterContext:



Public Member Functions

- **CounterContext** (const QVariantHash &map)
- virtual bool canEval (const QString &key) const
- virtual QString eval (const QString &key, const QString &_template, Mustache::Renderer *renderer)
- virtual QString stringValue (const QString &key) const
- **CounterContext** (const QVariantHash &map)
- virtual bool canEval (const QString &key) const
- virtual QString eval (const QString &key, const QString &_template, Mustache::Renderer *renderer)
- virtual QString stringValue (const QString &key) const

Public Attributes

- int **counter**

Additional Inherited Members

8.41.1 Member Function Documentation

8.41.1.1 virtual bool CounterContext::canEval (const QString & *key*) const [inline], [virtual]

Returns true if eval() should be used to render section tags using `key`. If canEval() returns true for a key, the renderer will pass the literal, unrendered block of text for the section to eval() and replace the section with the result.

canEval() and eval() are equivalents for callable objects (eg. lambdas) in other Mustache implementations.

The default implementation always returns false.

Reimplemented from Mustache::QtVariantContext.

8.41.1.2 virtual bool CounterContext::canEval (const QString & *key*) const [inline], [virtual]

Returns true if eval() should be used to render section tags using `key`. If canEval() returns true for a key, the renderer will pass the literal, unrendered block of text for the section to eval() and replace the section with the result.

canEval() and eval() are equivalents for callable objects (eg. lambdas) in other Mustache implementations.

The default implementation always returns false.

Reimplemented from Mustache::QtVariantContext.

8.41.1.3 virtual QString CounterContext::eval (const QString & *key,* const QString & *_template,* Mustache::Renderer * *renderer*) [inline], [virtual]

Callback used to render a template section with the given `key`. `renderer` will substitute the original section tag with the result of eval().

The default implementation returns an empty string.

Reimplemented from Mustache::QtVariantContext.

8.41.1.4 virtual QString CounterContext::eval (const QString & *key,* const QString & *_template,* Mustache::Renderer * *renderer*) [inline], [virtual]

Callback used to render a template section with the given `key`. `renderer` will substitute the original section tag with the result of eval().

The default implementation returns an empty string.

Reimplemented from Mustache::QtVariantContext.

8.41.1.5 virtual QString CounterContext::stringValue (const QString & *key*) const [inline], [virtual]

Returns a string representation of the value for `key` in the current context. This is used to replace a Mustache value tag.

Reimplemented from Mustache::QtVariantContext.

8.41.1.6 virtual QString CounterContext::stringValue (const QString & *key*) const [inline], [virtual]

Returns a string representation of the value for `key` in the current context. This is used to replace a Mustache value tag.

Reimplemented from Mustache::QtVariantContext.

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

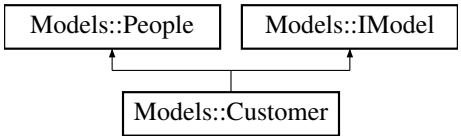
- src/libs/qt-mustache/tests/test_mustache.cpp

8.42 Models::Customer Class Reference

The Customer class Customer.

`#include <customer.h>`

Inheritance diagram for Models::Customer:



Public Member Functions

- Customer ()

  Customer::Customer Construct a Customer.
- Customer (int id)

Customer::Customer Constuct a Customer who is specidied by id

- void commit ()

  Customer::commit Update customer data on the database.
- void hydrat (int id)

  Customer::hydrat Insert into database informations related to the Customer who is specified by id
- void remove ()

  Customer::remove Remove the current customer.
- QVariantHash getDataMap ()

  getDataMap Get all data of model with a HashMap key/value
- QString getPath () const

  Customer::getPath Return the path of the workspace for the current Customer.
- QString getNameFolder () const

  Customer::getNameFolder Return the name of the current Customer's folder in the workspace.
- double getTurnover () const

  Customer::getTurnover Return the turnover of the customer money that customer pay, revenue sales.

Additional Inherited Members

8.42.1 Detailed Description

The Customer class Customer.

Author

Antoine de Roquemaurel
Florent Berbie

8.42.2 Constructor & Destructor Documentation

8.42.2.1 Models::Customer::Customer (int *id*)

Customer::Customer Constuct a Customer who is specidied by *id*

Parameters

id	Customer identify

8.42.3 Member Function Documentation

8.42.3.1 QVariantHash Models::Customer::getDataMap ( ) [virtual]

getDataMap Get all data of model with a HashMap key/value

Returns

Model's data

Implements Models::IModel.

8.42.3.2 QString Models::Customer::getNameFolder ( ) const

Customer::getNameFolder Return the name of the current Customer's folder in the workspace.

Returns

name of the Customer's folder

8.42.3.3 QString Models::Customer::getPath ( ) const

Customer::getPath Return the path of the workspace for the current Customer.

Returns

workspace path

8.42.3.4 double Models::Customer::getTurnover ( ) const

Customer::getTurnover Return the turnover of the customer money that customer pay, revenue sales.

Returns

turnover

8.42.3.5 void Models::Customer::hydrat (int *id*) [virtual]

Customer::hydrat Insert into database informations related to the Customer who is specified by *id*

Parameters

id	Customer identify

Implements Models::IModel.

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

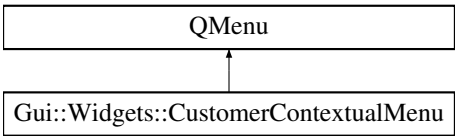
- src/models/customer.h
- src/models/customer.cpp

8.43 Gui::Widgets::CustomerContextualMenu Class Reference

Display contextual menu on a customer.

`#include <customercontextualmenu.h>`

Inheritance diagram for Gui::Widgets::CustomerContextualMenu:



Public Member Functions

- CustomerContextualMenu (QWidget ∗w=0)

  CustomerContextualMenu::CustomerContextualMenu Construct a new contextual menu.
- ∼CustomerContextualMenu ()

  CustomerContextualMenu::Destruct the contextual menu.

8.43.1 Detailed Description

Display contextual menu on a customer.

Author

Antoine de Roquemaurel

8.43.2 Constructor & Destructor Documentation

8.43.2.1 Gui::Widgets::CustomerContextualMenu::CustomerContextualMenu (QWidget * w = 0)

CustomerContextualMenu::CustomerContextualMenu Construct a new contextual menu.

Parameters

w	Parent

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

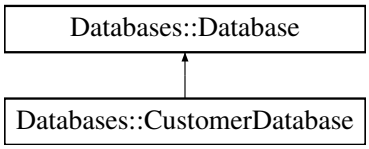
- src/gui/widgets/customercontextualmenu.h
- src/gui/widgets/customercontextualmenu.cpp

8.44 Databases::CustomerDatabase Class Reference

The **CustomerDatabase** class Customer table database.

`#include <customerdatabase.h>`

Inheritance diagram for Databases::CustomerDatabase:



Public Member Functions

- WdgModels::CustomersTableModel * getCustomersTable (QString filter="") throw (DbException*)

  CustomerDatabase::getCustomersTable Return an item model of customers for QTableView.
- QStandardItemModel * getTree (QString filter="") throw (DbException*)

  CustomerDatabase::getTree Return an item model of customers for QTree.
- QSharedPointer< Models::Customer > getCustomer (const int pId)

  CustomerDatabase::getCustomer get informations about the customer identified by pId
- int addCustomer (const Models::Customer &)

  CustomerDatabase::addCustomer Add the customer pCustomer to the database.
- void updateCustomer (Customer &)

  CustomerDatabase::updateCustomer Update informations about the customer pCustomer
- void removeCustomer (const int pId)

  CustomerDatabase::removeCustomer Remove the customer with the id pId
- int getNbCustomers ()

  CustomerDatabase::getNbCustomers Return the number of customers existing.
- QStandardItem * getItemRoot ()

CustomerDatabase::getItemRoot Return the first item for the QStandardItemModel.

- QStandardItem * getItemCustomer (QSqlQuery q1)

  CustomerDatabase::getItemCustomer Return the customer item for the QStandardItemModel.
- QStandardItem * getItemProject (QSqlQuery q2)

  CustomerDatabase::getItemProject Return the project item for the QStandardItemModel.
- QSharedPointer< Models::Customer > getCustomer (QSqlQuery &q)

  CustomerDatabase::getCustomer Add the element of the q request and return their.
- void updateCustomer (QSqlQuery &q, Customer &pCustomer)

  CustomerDatabase::updateCustomer Update customer data according to the request q

Static Public Member Functions

- static CustomerDatabase * instance () throw (DbException*)

  *CustomerDatabase::instance Return an instance of **CustomerDatabase***

Additional Inherited Members

8.44.1 Detailed Description

The **CustomerDatabase** class Customer table database.

Author

Antoine de Roquemaurel
Manantsoa Razanajatovo
Florent Berbie

See also

Database
Customer

8.44.2 Member Function Documentation

8.44.2.1 int Databases::CustomerDatabase::addCustomer (const Models::Customer & *pCustomer*)

CustomerDatabase::addCustomer Add the customer *pCustomer* to the database.

Returns

customer id

8.44.2.2 QSharedPointer< Models::Customer > Databases::CustomerDatabase::getCustomer (const int *pId*)

CustomerDatabase::getCustomer get informations about the customer identified by *pId*

Parameters

pId	customer id

Returns

the Customer

8.44.2.3 QSharedPointer< Models::Customer > Databases::CustomerDatabase::getCustomer (QSqlQuery & *q*)

CustomerDatabase::getCustomer Add the element of the *q* request and return their.

Parameters

q	SQL request

Returns

a customer formed according to QSharedPointer

8.44.2.4 WdgModels::CustomersTableModel * Databases::CustomerDatabase::getCustomersTable (QString *filter* = " ") throw DbException *)

CustomerDatabase::getCustomersTable Return an item model of customers for QTableView.

Parameters

filter	Select only customers who are specified by *filter*

Exceptions

DbException	

Returns

QStandardItemModel an item model for QTableView

8.44.2.5 QStandardItem * Databases::CustomerDatabase::getItemCustomer (QSqlQuery *q1*)

CustomerDatabase::getItemCustomer Return the customer item for the QStandardItemModel.

Parameters

q1	the row of the sql query for customers

Returns

QStandardItem an item for QTree (level/depth 1)

8.44.2.6 QStandardItem * Databases::CustomerDatabase::getItemProject (QSqlQuery *q2*)

CustomerDatabase::getItemProject Return the project item for the QStandardItemModel.

Parameters

q2	the row of the sql query for projects

Returns

QStandardItem an item for QTree (level/depth 2)

8.44.2.7 QStandardItem * Databases::CustomerDatabase::getItemRoot ( )

CustomerDatabase::getItemRoot Return the first item for the QStandardItemModel.

Returns

QStandardItem an item for QTree (level/depth 0)

8.44.2.8 int Databases::CustomerDatabase::getNbCustomers ( )

CustomerDatabase::getNbCustomers Return the number of customers existing.

Returns

number of customers

8.44.2.9 QStandardItemModel * Databases::CustomerDatabase::getTree (QString *filter* = " ") throw DbException *)

CustomerDatabase::getTree Return an item model of customers for QTree.

Parameters

filter	Select only customers who are specified by *filter*

Exceptions

DbException	

Returns

QStandardItemModel an item model for QTreeView

8.44.2.10 CustomerDatabase * Databases::CustomerDatabase::instance ( ) throw DbException *) [static]

CustomerDatabase::instance Return an instance of **CustomerDatabase**

See also

DbException

Returns

Instance of CustomerDatabase

8.44.2.11 void Databases::CustomerDatabase::removeCustomer (const int *pId*)

CustomerDatabase::removeCustomer Remove the customer with the id *pId*

Parameters

pId	customer id

8.44.2.12 void Databases::CustomerDatabase::updateCustomer (QSqlQuery & *q*, Customer & *pCustomer*)

CustomerDatabase::updateCustomer Update customer data according to the request *q*

Parameters

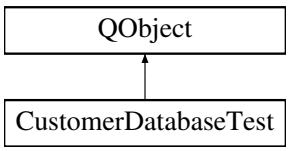
q	SQL request

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

- src/database/customerdatabase.h
- src/database/customerdatabase.cpp

8.45 CustomerDatabaseTest Class Reference

Inheritance diagram for CustomerDatabaseTest:



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

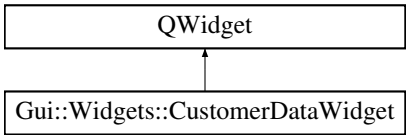
- tests/database/customerdatabasetest.h
- tests/database/customerdatabasetest.cpp

8.46 Gui::Widgets::CustomerDataWidget Class Reference

Class for display info of a customer.

`#include <customerdatawidget.h>`

Inheritance diagram for Gui::Widgets::CustomerDataWidget:



Public Member Functions

- CustomerDataWidget (QWidget *parent=0)

  CustomerDataWidget::CustomerDataWidget Contruct a CustomerDataWidget.
- void printUserData ()

  CustomerDataWidget::printUserData Print Data of current user.
- void printInformations (int id)

  CustomerDataWidget::printInformations Print Data of customer id.

8.46.1 Detailed Description

Class for display info of a customer.

Author

8.46.2 Constructor & Destructor Documentation

8.46.2.1 Gui::Widgets::CustomerDataWidget::CustomerDataWidget (QWidget * *parent* = `0`) [explicit]

CustomerDataWidget::CustomerDataWidget Contruct a CustomerDataWidget.

Parameters

parent	Widget parent

8.46.3 Member Function Documentation

8.46.3.1 void Gui::Widgets::CustomerDataWidget::printInformations (int *id*)

CustomerDataWidget::printInformations Print Data of customer id.

Parameters

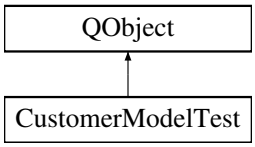
id	of customer to print

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

- src/gui/widgets/customerdatawidget.h
- src/gui/widgets/customerdatawidget.cpp

8.47 CustomerModelTest Class Reference

Inheritance diagram for CustomerModelTest:



Public Member Functions

- void **setup** ()

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

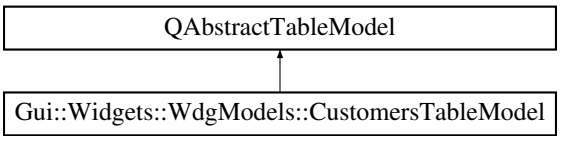
- tests/models/customermodeltest.h
- tests/models/customermodeltest.cpp

8.48 Gui::Widgets::WdgModels::CustomersTableModel Class Reference

The CustomersTableModel class for a customer table.

`#include <customerstablemodel.h>`

Inheritance diagram for Gui::Widgets::WdgModels::CustomersTableModel:



Public Member Functions

- CustomersTableModel (QObject *parent=0)

  CustomersTableModel::CustomersTableModel Construct a CustomersTableModel.
- int rowCount (const QModelIndex &) const

  CustomersTableModel::rowCount Number of customers row.
- int columnCount (const QModelIndex &) const

  CustomersTableModel::columnCount Number of column of a customer.
- QVariant data (const QModelIndex &index, int role=Qt::DisplayRole) const

  CustomersTableModel::data Obtains data of a specify cell.
- QVariant headerData (int section, Qt::Orientation orientation, int role=Qt::DisplayRole) const

  CustomersTableModel::headerData Obtains header title of table.
- bool setData (const QModelIndex &index, const QVariant &value, int role=Qt::EditRole)

  CustomersTableModel::setData Change data of a cell.
- void append (const Customer &customer)

  CustomersTableModel::append Add a new line in table.
- void remove (const int i)

  CustomersTableModel::remove Remove a line.
- Qt::ItemFlags flags (const QModelIndex &index) const

  CustomersTableModel::flags Differents table flags.
- int count ()

  CustomersTableModel::count Number of customers in table.
- QList< Customer > getCustomers () const

  CustomersTableModel::getCustomers Return the list of customers.

8.48.1 Detailed Description

The CustomersTableModel class for a customer table.

Author

Florent Berbie

See also

Customer

8.48.2 Constructor & Destructor Documentation

8.48.2.1 Gui::Widgets::WdgModels::CustomersTableModel::CustomersTableModel (QObject * *parent* = 0)

CustomersTableModel::CustomersTableModel Construct a CustomersTableModel.

Parameters

parent	Parent widget

8.48.3 Member Function Documentation

8.48.3.1 void Gui::Widgets::WdgModels::CustomersTableModel::append (const Customer & *customer*)

CustomersTableModel::append Add a new line in table.

Parameters

Customer	The new customer

8.48.3.2 int Gui::Widgets::WdgModels::CustomersTableModel::columnCount (const QModelIndex &) const

CustomersTableModel::columnCount Number of column of a customer.

Returns

The number of column

8.48.3.3 int Gui::Widgets::WdgModels::CustomersTableModel::count ( )

CustomersTableModel::count Number of customers in table.

Returns

The number of customers

8.48.3.4 QVariant Gui::Widgets::WdgModels::CustomersTableModel::data (const QModelIndex & *index*, int *role* = `Qt::DisplayRole`) const

CustomersTableModel::data Obtains data of a specify cell.

Parameters

index	The cell who we want data
role	The role of set

Returns

The data of cell

8.48.3.5 Qt::ItemFlags Gui::Widgets::WdgModels::CustomersTableModel::flags (const QModelIndex & *index*) const

CustomersTableModel::flags Differents table flags.

Parameters

index	The cell who we want to know flags

Returns

Flags

8.48.3.6 QList< Customer > Gui::Widgets::WdgModels::CustomersTableModel::getCustomers ( ) const

CustomersTableModel::getCustomers Return the list of customers.

Returns

list of Customers

8.48.3.7 QVariant Gui::Widgets::WdgModels::CustomersTableModel::headerData (int *section,* Qt::Orientation *orientation,* int *role* = `Qt::DisplayRole`) const

CustomersTableModel::headerData Obtains header title of table.

Parameters

section	The number of column
orientation	The table orientation
role	

Returns

The Title header of column

8.48.3.8 void Gui::Widgets::WdgModels::CustomersTableModel::remove (const int *i*)

CustomersTableModel::remove Remove a line.

Parameters

i	The number of line to remove

8.48.3.9 int Gui::Widgets::WdgModels::CustomersTableModel::rowCount (const QModelIndex &) const

CustomersTableModel::rowCount Number of customers row.

Returns

The number of customers

8.48.3.10 bool Gui::Widgets::WdgModels::CustomersTableModel::setData (const QModelIndex & *index*, const QVariant & *value*, int *role* = `Qt::EditRole`)

CustomersTableModel::setData Change data of a cell.

Parameters

index	The cell to change data
value	The new value
role	The role of cell

Returns

True if we could edit

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

- src/gui/widgets/widgetsmodels/customerstablemodel.h
- src/gui/widgets/widgetsmodels/customerstablemodel.cpp

8.49 Databases::Database Class Reference

The **Database** class Master class for all database access.

`#include <database.h>`

Inheritance diagram for Databases::Database:

Public Member Functions

- QString lastError (const QSqlQuery &q) const

  Database::lastError Return an error message on the last error occured during the SQL request q
- void testCases ()

  Database::testCases Realise a test cases.
- void executeFile (QString pName)

  Database::executerFichier Exeute a specified file named pName
- void openTransaction ()

  Database::openTransaction Open new transaction.
- void closeTransaction ()

  Database::closeTransaction Close current transaction.
- void close ()

  Database::close Close database access.
- void open ()

  Database::open Open database.
- ~Database ()

  Database::~Database Suppression object, and close database access.
- void setDatabase (QSqlDatabase sql)

  Database::setDatabase Set database.
- void updateBillingNumber ()

  Database::updateBillingNumber Update the billing number.
- void cleanDatabase ()

  Database::clearDatabase Drop alls tables of Database WARNING: We can't restore data after.
- void changeDatabase (Databases::DbType dbType)

  changeDatabase Change the current database : mysql to sqlite or sqlite to mysql

Static Public Member Functions

- static Database * instance (bool tests=false) throw (DbException*)

  Database::getInstance Return an instance of Database.

Protected Member Functions

- Database (bool tests=false) throw (DbException*)

  Database::Database Database is a singleton.
- QVariant value (const QSqlQuery &q, const QString &champ) const

  Database::valeur Value of database field.

Protected Attributes

- QSettings * _settings

  settings
- QSqlDatabase mDatabase

  contains Database
- QList< Database * > _instances

  List of instances.

Static Protected Attributes

- static Database * _instance = 0

  Instance.
- static bool _dbInstance = 0

  an instance of db is open
- static bool isOpen = false

  Database is open.
- static bool **_isMysql** = false

8.49.1 Detailed Description

The **Database** class Master class for all database access.

Author

Antoine de Roquemaurel

8.49.2 Member Function Documentation

8.49.2.1 void Databases::Database::changeDatabase (Databases::DbType *dbType*)

changeDatabase Change the current database : mysql to sqlite or sqlite to mysql

Parameters

dbType	: The new database type, Sqlite or Mysql

8.49.2.2 void Databases::Database::executeFile (QString *pName*)

Database::executerFichier Exeute a specified file named *pName*

Parameters

pNom	File name

8.49.2.3 Database * Databases::Database::instance (bool *tests* = `false`) throw DbException *) [static]

Database::getInstance Return an instance of Database.

Returns

Instance of Database

8.49.2.4 QString Databases::Database::lastError (const QSqlQuery & *q*) const [inline]

Database::lastError Return an error message on the last error occured during the SQL request *q*

Parameters

q	SQL request

Returns

an error message

8.49.2.5 void Databases::Database::setDatabase (QSqlDatabase *sql*)

Database::setDatabase Set database.

Parameters

sql	The new database

8.49.2.6 QVariant Databases::Database::value (const QSqlQuery & *q*, const QString & *champ*) const [protected]

Database::valeur Value of database field.

Parameters

q	Query
champ	Field

Returns

The value

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

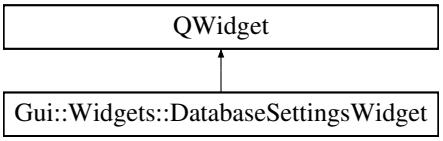
- src/database/database.h
- src/database/database.cpp

8.50 Gui::Widgets::DatabaseSettingsWidget Class Reference

The DatabaseSettingsWidget class Windows of database settings.

`#include <databasesettingswidget.h>`

Inheritance diagram for Gui::Widgets::DatabaseSettingsWidget:



Public Slots

- bool isValid ()

  DatabaseSettingsWidget::isValid Return TRUE if all fields are correctly inputed else FALSE.
- void checkRepeatPassword (QString text)

  DatabaseSettingsWidget::checkRepeatLogin Check if the second login field is the same than the first.
- void userInterfaceChanged ()

  DatabaseSettingsWidget::userInterfaceChanged User interface has changed.

Signals

- void textfieldChanged ()

  DatabaseSettingsWidget::textfieldChanged Signal which indicates if a fieldtext has changed.

Public Member Functions

- DatabaseSettingsWidget (QWidget *parent=0)

  DatabaseSettingsWidget::DatabaseSettingsWidget Construct a DatabaseSettingsWidget.
- void fillFields ()

  DatabaseSettingsWidget::fillFields Complete fields with a default value for field Database name, Username, IP address and port.
- QString getDatabaseName ()

  DatabaseSettingsWidget::getDatabaseName Return the database name.
- QString getLogin ()

  DatabaseSettingsWidget::getLogin Return the login of the user.
- QString getPassword ()

  DatabaseSettingsWidget::getPassword Return the password of the user.
- QString getDomainNameOrIP ()

  DatabaseSettingsWidget::getDomainNameOrIP Return the domain name or IP address on the machine where is the database.
- QString getPort ()

  DatabaseSettingsWidget::getPort Return the port of the database.

8.50.1 Detailed Description

The DatabaseSettingsWidget class Windows of database settings.

Author

8.50.2 Constructor & Destructor Documentation

8.50.2.1 Gui::Widgets::DatabaseSettingsWidget::DatabaseSettingsWidget (QWidget * *parent* = 0) [explicit]

DatabaseSettingsWidget::DatabaseSettingsWidget Construct a DatabaseSettingsWidget.

Parameters

parent	Parent widget of this windows

8.50.3 Member Function Documentation

8.50.3.1 QString Gui::Widgets::DatabaseSettingsWidget::getDatabaseName ( )

DatabaseSettingsWidget::getDatabaseName Return the database name.

Returns

Database name

8.50.3.2 QString Gui::Widgets::DatabaseSettingsWidget::getDomainNameOrIP ( )

DatabaseSettingsWidget::getDomainNameOrIP Return the domain name or IP address on the machine where is the database.

Returns

Domain name or IP address

8.50.3.3 QString Gui::Widgets::DatabaseSettingsWidget::getLogin ( )

DatabaseSettingsWidget::getLogin Return the login of the user.

Returns

Login of the user

8.50.3.4 QString Gui::Widgets::DatabaseSettingsWidget::getPassword ( )

DatabaseSettingsWidget::getPassword Return the password of the user.

Returns

User password

8.50.3.5 QString Gui::Widgets::DatabaseSettingsWidget::getPort ( )

DatabaseSettingsWidget::getPort Return the port of the database.

Returns

Database port

8.50.3.6 bool Gui::Widgets::DatabaseSettingsWidget::isValid ( ) [slot]

DatabaseSettingsWidget::isValid Return TRUE if all fields are correctly inputed else FALSE.

Returns

boolean

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

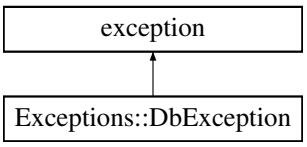
- src/gui/widgets/databasesettingswidget.h
- src/gui/widgets/databasesettingswidget.cpp

8.51 Exceptions::DbException Class Reference

The DbException class for database exception : queries, db file, . . .

`#include <dbexception.h>`

Inheritance diagram for Exceptions::DbException:



Public Member Functions

- DbException (const QString fct, const QString fctName, const QString logError, float errorCode)

  DbException::DbException. Construct a DbException.
- virtual ~DbException () throw ()

  ~DbException
- void popupMessage (QWidget *parent)

  DbException::popupMessage. Display a popup message with the message error.

8.51.1 Detailed Description

The DbException class for database exception : queries, db file, . . .

Author

Antoine de Roquemaurel

8.51.2 Constructor & Destructor Documentation

8.51.2.1 Exceptions::DbException::DbException (const QString *fct*, const QString *fctName*, const QString *logError*, float *errorCode*)

DbException::DbException. Construct a DbException.

Parameters

userError	ClassName of error
fctName	Function name
logError	Message error
errorCode	Code of error

8.51.3 Member Function Documentation

8.51.3.1 void Exceptions::DbException::popupMessage (QWidget * *parent*)

DbException::popupMessage. Display a popup message with the message error.

Parameters

parent	

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

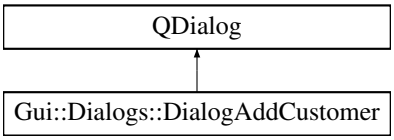
- src/exceptions/dbexception.h
- src/exceptions/dbexception.cpp

8.52 Gui::Dialogs::DialogAddCustomer Class Reference

The DialogAddCustomer class Window to add or modify a Customer.

`#include <dialogaddcustomer.h>`

Inheritance diagram for Gui::Dialogs::DialogAddCustomer:



Public Slots

- void checkFields ()

  DialogAddCustomer::checkFields Check if fields are valid.

Public Member Functions

- DialogAddCustomer (int id=0, QWidget ∗parent=0)

  DialogAddCustomer::DialogAddCustomer Construct a window to add/modify a Customer.
- void fillFields ()

  DialogAddCustomer::fillFields If the Customer exits, fill line edits with the data of the current Customer.
- void accept ()

  DialogAddCustomer::accept Valid data inputed by user and add these data in Database.
- void reject ()

  DialogAddCustomer::reject Cancel the operation and close the windows.

8.52.1 Detailed Description

The DialogAddCustomer class Window to add or modify a Customer.

Author

8.52.2 Constructor & Destructor Documentation

8.52.2.1 Gui::Dialogs::DialogAddCustomer::DialogAddCustomer (int *id* = 0, QWidget ∗ *parent* = 0) [explicit]

DialogAddCustomer::DialogAddCustomer Construct a window to add/modify a Customer.

Parameters

id	Customer id
parent	QWidget parent

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

- src/gui/dialogs/dialogaddcustomer.h
- src/gui/dialogs/dialogaddcustomer.cpp

8.53 Utils::Directories Class Reference

Static Public Member Functions

- static QString makeDirectory (QDir &directory, const QString path, const QString folder) throw (Exceptions↩ ::FileException∗)

  MainWindow::makeDirectory If not exists make a new directory folder

8.53.1 Member Function Documentation

8.53.1.1 QString Utils::Directories::makeDirectory (QDir & *directory,* const QString *path,* const QString *folder*) throw Exceptions::FileException ∗) [static]

MainWindow::makeDirectory If not exists make a new directory *folder*

Parameters

path	Return the path of the folder just created
folder	Folder name to create

Returns

Path of the folder just created

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

- src/utils/directories.h
- src/utils/directories.cpp

8.54 Utils::Double Class Reference

The Double class Utils functions for Double calculs.

`#include <double.h>`

Static Public Member Functions

- static double round (double n, unsigned int d)

  round Roud a double value to d decimals
- static double **round** (double n, unsigned int d)

8.54.1 Detailed Description

The Double class Utils functions for Double calculs.

8.54.2 Member Function Documentation

8.54.2.1 double Utils::Double::round (double *n,* unsigned int *d*) [static]

round Roud a double value to d decimals

Parameters

n	The number
d	The number of decimals who you want

Returns

The rounded value

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

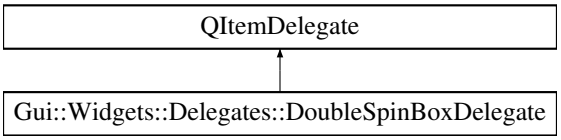
- src/utils/double.h
- src/utils/double.cpp

8.55 Gui::Widgets::Delegates::DoubleSpinBoxDelegate Class Reference

The DoubleSpinBoxDelegate class.

`#include <doublespinboxdelegate.h>`

Inheritance diagram for Gui::Widgets::Delegates::DoubleSpinBoxDelegate:



Public Member Functions

- DoubleSpinBoxDelegate (QObject ∗parent=0)

  DoubleSpinBoxDelegate::DoubleSpinBoxDelegate.
- QWidget ∗ createEditor (QWidget ∗parent, const QStyleOptionViewItem &option, const QModelIndex &index) const

  DoubleSpinBoxDelegate::createEditor Return a ComboBox specified by index item defined by the parent widget and style option which are used to control how the editor widgets appears.
- void setEditorData (QWidget ∗editor, const QModelIndex &index) const

  DoubleSpinBoxDelegate::setEditorData Sets the data to be displayed and edited by the editor from the data model item specified by the model index
- void setModelData (QWidget ∗editor, QAbstractItemModel ∗model, const QModelIndex &index) const

  DoubleSpinBoxDelegate::setEditorData Sets the data to be displayed and edited by the editor from the data model item specified by the model index
- void updateEditorGeometry (QWidget ∗editor, const QStyleOptionViewItem &option, const QModelIndex &index) const

  DoubleSpinBoxDelegate::updateEditorGeometry Update the editor for the item specified by index according to the style option given.

8.55.1 Detailed Description

The DoubleSpinBoxDelegate class.

Author

Florent Berbie

8.55.2 Constructor & Destructor Documentation

8.55.2.1 Gui::Widgets::Delegates::DoubleSpinBoxDelegate::DoubleSpinBoxDelegate (QObject * *parent* = 0)

DoubleSpinBoxDelegate::DoubleSpinBoxDelegate.

Parameters

parent	

8.55.3 Member Function Documentation

8.55.3.1 QWidget * Gui::Widgets::Delegates::DoubleSpinBoxDelegate::createEditor (QWidget * *parent*, const QStyleOptionViewItem & *option*, const QModelIndex & *index*) const

DoubleSpinBoxDelegate::createEditor Return a ComboBox specified by *index* item defined by the *parent* widget and style *option* which are used to control how the editor widgets appears.

Parameters

parent	Widget parent
option	Option style
index	Index for editing

Returns

DoubleSpinBoxDelegate

8.55.3.2 void Gui::Widgets::Delegates::DoubleSpinBoxDelegate::setEditorData (QWidget * *editor*, const QModelIndex & *index*) const

DoubleSpinBoxDelegate::setEditorData Sets the data to be displayed and edited by the *editor* from the data model item specified by the model *index*

Parameters

editor	Data edited
index	Index of the model to edit

8.55.3.3 void Gui::Widgets::Delegates::DoubleSpinBoxDelegate::setModelData (QWidget * *editor*, QAbstractItemModel * *model*, const QModelIndex & *index*) const

DoubleSpinBoxDelegate::setEditorData Sets the data to be displayed and edited by the *editor* from the data model item specified by the model *index*

Parameters

editor	Data edited
index	Index of the model to edit

8.55.3.4 void Gui::Widgets::Delegates::DoubleSpinBoxDelegate::updateEditorGeometry (QWidget * *editor*, const QStyleOptionViewItem & *option*, const QModelIndex & *index*) const

DoubleSpinBoxDelegate::updateEditorGeometry Update the *editor* for the item specified by *index* according to the style *option* given.

Parameters

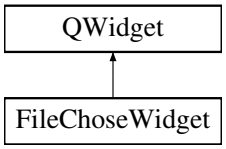
editor	Editor widget to update
option	Style option
index	Item index

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

- src/gui/widgets/delegates/doublespinboxdelegate.h
- src/gui/widgets/delegates/doublespinboxdelegate.cpp

8.56 FileChoseWidget Class Reference

Inheritance diagram for FileChoseWidget:



Public Member Functions

- **FileChoseWidget** (QWidget ∗parent=0)

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

- src/gui/widgets/filechosewidget.h
- src/gui/widgets/filechosewidget.cpp

8.57 Exceptions::FileException Class Reference

The FileException class for file/acess file exception.

`#include <fileexception.h>`

Public Member Functions

- FileException (const QString userError, const QString fctName, const QString logError, float errorCode)

  FileException::FileException. Construct a FileException.
- void popupMessage (QWidget ∗parent)

  FileException::popupMessage. Display a popup message with the message error.

8.57.1 Detailed Description

The FileException class for file/acess file exception.

Author

Florent Berbie

8.57.2 Constructor & Destructor Documentation

8.57.2.1 Exceptions::FileException::FileException (const QString *userError,* const QString *fctName,* const QString *logError,* float *errorCode*)

FileException::FileException. Construct a FileException.

Parameters

userError	ClassName of error
fctName	Function name
logError	Message error
errorCode	Code of error

8.57.3 Member Function Documentation

8.57.3.1 void Exceptions::FileException::popupMessage (QWidget * *parent*)

FileException::popupMessage. Display a popup message with the message error.

Parameters

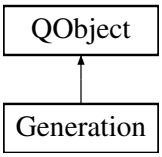
parent	

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

- src/exceptions/fileexception.h
- src/exceptions/fileexception.cpp

8.58 Generation Class Reference

Inheritance diagram for Generation:



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

- tests/generation.h
- tests/generation.cpp

8.59 Utils::HierarchicalSystem Class Reference

The HierarchicalSystem class Create class which contains hierarchical system of FactDev.

`#include <hierarchicalsystem.h>`

Public Member Functions

- HierarchicalSystem ()

  HierarchicalSystem::HierarchicalSystem Construct a HierarchicalSystem.
- void getAllProjects ()

  HierarchicalSystem::getAllProjects Get all projects and add each project to Customer linked.
- void getAllBillings ()

  HierarchicalSystem::getAllBillings Get all billings and add each billing to Project linked.
- void updateData ()

HierarchicalSystem::updateData *Update data on Customers, Projects, Billings.*

- void addProjectToCustomer (Project ∗p, Customer c)

  HierarchicalSystem::addProjectToCustomer *Add the Project p to the Customer c*

- void addBillingToProject (Billing ∗b, Project ∗p)

  HierarchicalSystem::addBillingToProject *Add the Billing b to the Project p*

- QMap< Project ∗, Customer > getCustomers () const

  HierarchicalSystem::getCustomers *Return all customers and these projects linked.*

- QMap< Billing ∗, Project ∗ > getProjects () const

  HierarchicalSystem::getProjects *Return all projects and these billing linked.*

8.59.1 Detailed Description

The HierarchicalSystem class Create class which contains hierarchical system of FactDev.

Author

Florent Berbie

See also

Customer
Project
Billing

8.59.2 Member Function Documentation

8.59.2.1 void Utils::HierarchicalSystem::addBillingToProject (Billing ∗ *b*, Project ∗ *p*)

HierarchicalSystem::addBillingToProject Add the Billing *b* to the Project p

Parameters

b	Billing
p	Project

8.59.2.2 void Utils::HierarchicalSystem::addProjectToCustomer (Project ∗ *p*, Customer *c*)

HierarchicalSystem::addProjectToCustomer Add the Project *p* to the Customer *c*

Parameters

p	Project
c	Customer

8.59.2.3 QMap< Project ∗, Customer > Utils::HierarchicalSystem::getCustomers ( ) const

HierarchicalSystem::getCustomers Return all customers and these projects linked.

Returns

Projects linked to Customers

8.59.2.4 QMap< Billing *, Project * > Utils::HierarchicalSystem::getProjects (void) const

HierarchicalSystem::getProjects Return all projects and these billing linked.

Returns

Billing linked to Projects

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

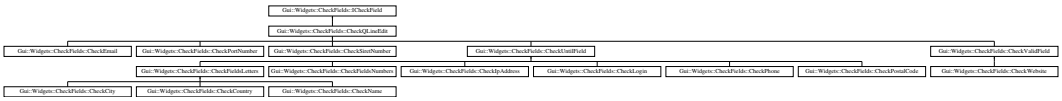
- src/utils/hierarchicalsystem.h
- src/utils/hierarchicalsystem.cpp

8.60 Gui::Widgets::CheckFields::ICheckField Class Reference

The ICheckField class Interface to check fields validity.

`#include <icheckfield.h>`

Inheritance diagram for Gui::Widgets::CheckFields::ICheckField:



Public Member Functions

- virtual bool check (QString text)=0

  ICheckField::check Check if the field (line edit) is valid Return TRUE if the field is valid, else FALSE.

8.60.1 Detailed Description

The ICheckField class Interface to check fields validity.

8.60.2 Member Function Documentation

8.60.2.1 virtual bool Gui::Widgets::CheckFields::ICheckField::check (QString *text*) [pure virtual]

ICheckField::check Check if the field (line edit) is valid Return TRUE if the field is valid, else FALSE.

Returns

boolean Validity of the text

Implemented in Gui::Widgets::CheckFields::CheckEmail, Gui::Widgets::CheckFields::CheckPortNumber, Gui::Widgets::CheckFields::CheckFieldsNumbers, Gui::Widgets::CheckFields::CheckIpAddress, Gui::Widgets::CheckFields::CheckWebsite, Gui::Widgets::CheckFields::CheckLogin, Gui::Widgets::CheckFields::CheckPhone, Gui::Widgets::CheckFields::CheckPostalCode, Gui::Widgets::CheckFields::CheckSiretNumber, Gui::Widgets::CheckFields::CheckUntilField, Gui::Widgets::CheckFields::CheckFieldsLetters, and Gui::Widgets::CheckFields::CheckValidField.

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

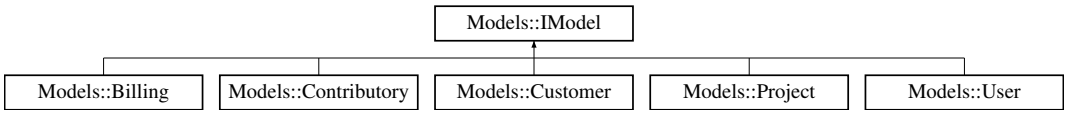
- src/gui/widgets/checkfields/icheckfield.h

8.61 Models::IModel Class Reference

The IModel class.

`#include <imodel.h>`

Inheritance diagram for Models::IModel:



Public Member Functions

- virtual ~IModel ()

  ~IModel Remove an instance of IModel

- virtual void commit ()=0

  IModel::commit Update or insert data into the database.

- virtual void hydrat (int id)=0

  IModel::hydrat Get data of the element which is specified by the identify id from the database.

- virtual void remove ()=0

  IModel::remove Remove the current element in the database.

- virtual QVariantHash getDataMap ()=0

  getDataMap Get all data of model with a HashMap key/value

- int getId () const

  IModel::getId Return the identify of the element of the database.

- void setId (int id)

  IModel::setId Replace the current identify by id

- bool isToRemoved () const

  toRemoved return if object must be removed.

- void setToRemoved (bool toRemoved)

  setToRemoved Change the flag for removed object

Protected Attributes

- int _id

  Element identify.

- bool _toRemoved

  Flag to know if the object must be removed.

8.61.1 Detailed Description

The IModel class.

Author

Antoine de Roquemaurel

8.61.2 Member Function Documentation

8.61.2.1 virtual QVariantHash Models::IModel::getDataMap ( ) [pure virtual]

getDataMap Get all data of model with a HashMap key/value

Returns

Model's data

Implemented in Models::Contributory, Models::Billing, Models::Project, Models::User, and Models::Customer.

8.61.2.2 int Models::IModel::getId ( ) const [inline]

IModel::getId Return the identify of the element of the database.

Returns

identity

8.61.2.3 virtual void Models::IModel::hydrat (int *id*) [pure virtual]

IModel::hydrat Get data of the element which is specified by the identify *id* from the database.

Parameters

id	

Implemented in Models::Billing, Models::Project, Models::Contributory, Models::User, and Models::Customer.

8.61.2.4 bool Models::IModel::isToRemoved ( ) const [inline]

toRemoved return if object must be removed.

Returns

boolean

8.61.2.5 void Models::IModel::setId (int *id*) [inline]

IModel::setId Replace the current identify by *id*

Parameters

id	New identify

8.61.2.6 void Models::IModel::setToRemoved (bool *toRemoved*) [inline]

setToRemoved Change the flag for removed object

Parameters

toRemoved	The new flag

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

- src/models/imodel.h

8.62 Utils::ItemType Class Reference

The ItemType class Item type model.

`#include <itemtype.h>`

Public Member Functions

- ItemType (int type, QString name)

  ItemType::ItemType Construct an Item type.
- QString getName () const

  ItemType::getName Get item name.
- Models::IModel * getModel (int id)

  *ItemType::getModel Get the databasemodel of the **ItemType** according to this identity id*
- void setName (const QString &name)

  ItemType::setName Modify the item name.
- int getType () const

  ItemType::getType Get the type of the current item.
- void setType (int type)

  ItemType::setType Modify the type of the current item.

Static Public Attributes

- static const int CUSTOMER = 0

  constant value assigned to Customer
- static const int PROJECT = 1

  constant value assigned to Project
- static const int BILLING = 2

  constant value assigned to Billing
- static const int QUOTE = 3

  constant value assigned to Quote

8.62.1 Detailed Description

The ItemType class Item type model.

8.62.2 Constructor & Destructor Documentation

8.62.2.1 Utils::ItemType::ItemType (int *type*, QString *name*)

ItemType::ItemType Construct an Item type.

Parameters

type	Type of the item
name	Name of the item

8.62.3 Member Function Documentation

8.62.3.1 Models::IModel * Utils::ItemType::getModel (int *id*)

ItemType::getModel Get the databasemodel of the **ItemType** according to this identity *id*

Parameters

id	Item type identity

Returns

database model

8.62.3.2 QString Utils::ItemType::getName ( ) const

ItemType::getName Get item name.

Returns

item name

8.62.3.3 int Utils::ItemType::getType ( ) const

ItemType::getType Get the type of the current item.

Returns

type of the current item

8.62.3.4 void Utils::ItemType::setName (const QString & *name*)

ItemType::setName Modify the item name.

Parameters

name	New Item name

8.62.3.5 void Utils::ItemType::setType (int *type*)

ItemType::setType Modify the type of the current item.

Parameters

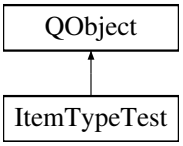
type	New item type

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

- src/utils/itemtype.h
- src/utils/itemtype.cpp

8.63 ItemTypeTest Class Reference

Inheritance diagram for ItemTypeTest:



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

- tests/utils/itemtypetest.h
- tests/utils/itemtypetest.cpp

8.64 Utils::Log Class Reference

The Log class for Simple management of log.

`#include <log.h>`

Public Member Functions

- ~Log ()

  Log::~Log.
- void write (const QString text)

  Log::write. Write log message in file.
- Log ()

  Log::Log. Log is a singleton.

Static Public Member Functions

- static Log & instance (TypeLog type=INFO)

  Log::instance. Return the instance of logger.

Friends

- Log & operator<< (Log &logger, const QString &text)

  operator << for log writing

8.64.1 Detailed Description

The Log class for Simple management of log.

8.64.2 Member Function Documentation

8.64.2.1 Log & Utils::Log::instance (TypeLog *type* = INFO) [static]

Log::instance. Return the instance of logger.

Parameters

type	Type of log : WARNING, INFO, ERROR

Returns

Instance of logger.

8.64.2.2 void Utils::Log::write (const QString *text*)

Log::write. Write log message in file.

Parameters

text	

8.64.3 Friends And Related Function Documentation

8.64.3.1 Log& operator<< (Log & *logger,* const QString & *text*) [friend]

operator << for log writing

Parameters

logger	Instance of Logger
text	Text to write

Returns

New logger.

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

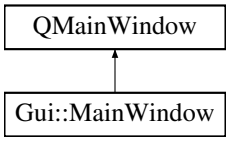
- src/utils/log.h
- src/utils/log.cpp

8.65 Gui::MainWindow Class Reference

The MainWindow class Main Window of the software.

`#include <mainwindow.h>`

Inheritance diagram for Gui::MainWindow:



Public Slots

- void addCustomer ()

  MainWindow::addCustomer open window to add a new customer.
- void editCustomer ()

MainWindow::editCustomer *open window to modify a customer.*

- void removeCustomer ()

  MainWindow::removeCustomer *open a popup to confirm the deletion of a customer, if ok remove the customer.*
- void addQuote ()

  MainWindow::addQuote *open window to add a new quote.*
- void addBill ()

  MainWindow::addBill *open window to add a new bill.*
- void billingIsPaid ()

  MainWindow::billingIsPaid *Define the current billing as "paid".*
- void editUser ()

  MainWindow::editUser *modify the user.*
- void search (QString s)

  MainWindow::search *launch a new search.*
- void addProject ()

  MainWindow::addProject *Create a new project for a customer.*
- void removeProject (void)

  MainWindow::removeProject *Remove a project for a customer.*
- void editProject (void)

  MainWindow::editProject *Modify the customer project.*
- void aboutQt ()

  MainWindow::aboutQt *show Qt's details.*
- void aboutFact ()

  MainWindow::aboutFact *show FACT's details (FACT team)*
- void aboutFactDev ()

  MainWindow::aboutFactDev() *show FactDev's details (FactDev Software)*
- void aboutIcons ()

  MainWindow::aboutIcons() *show icons's details.*
- void updateButtons (void)

  updateButton Update all button to disable or enabled its
- void editDoc ()

  MainWindow::editDoc *Edit the quote or bill of the project.*
- void removeDoc ()

  MainWindow::removeDoc *Remove the quote or bill of the project.*
- void copyDoc ()

  MainWindow::copyDoc *Copy all elements of a quote or a bill and Display these elements in a new quote or bill.*
- void openPdf ()

  MainWindow::openPdf *Open the PDF file of the current Quote or Billing selected in the TableView.*
- void computeTurnover ()

  MainWindow::computeTurnover *open window to allow computation of a period turnover.*

Public Member Functions

- MainWindow (QWidget *parent=0)

  MainWindow: Construct a window.
- int getCurrentCustomerId ()

  MainWindow::getCurrentCustomerId *get the selected customer.*
- int getCurrentProjectId ()

  MainWindow::getCurrentProjectId *get the selected project id.*
- int getCurrentQuoteId ()

  MainWindow::getCurrentQuoteId *get the selected quote id.*

- QString getCurrentCustomerName ()

  MainWindow::getCurrentCustomerName get the selected customer name in the customers' table.
- QString getCurrentProjectName ()

  MainWindow::getCurrentProjectName get the selected project name in the table of projects.
- int treeLevel ()

  MainWindow::treeLevel return the level of the node selected in the tree.
- QModelIndex rootTree ()

  MainWindow::rootTree return the root of the tree "Tous les clients".
- void addDoc (bool isBilling)

  MainWindow::addDoc open window to add a new document.
- void resizeEvent (QResizeEvent ∗event)

  *MainWindow::resizeEvent Resize central TableView when you resize the **MainWindow***
- void responsiveCustomerTable ()

  MainWindow::responsiveCustomerTable Resize the Customer TableView according it resolution.
- void responsiveProjectTable ()

  MainWindow::responsiveProjectTable Resize the Project TableView according it resolution.
- void responsiveBillingTable ()

  MainWindow::responsiveBillingTable Resize the Billing TableView according it resolution.

8.65.1 Detailed Description

The MainWindow class Main Window of the software.

Author

Everybody

8.65.2 Constructor & Destructor Documentation

8.65.2.1 Gui::MainWindow::MainWindow (QWidget ∗ *parent* = 0) [explicit]

MainWindow: Construct a window.

Parameters

parent	

8.65.3 Member Function Documentation

8.65.3.1 void Gui::MainWindow::addBill ( ) [slot]

MainWindow::addBill open window to add a new bill.

See also

AddQuoteDialog

8.65.3.2 void Gui::MainWindow::addDoc (bool *isBilling*)

MainWindow::addDoc open window to add a new document.

Parameters

bool	quote or bill

See also

addBill addQuote

8.65.3.3 void Gui::MainWindow::addProject ( ) [slot]

MainWindow::addProject Create a new project for a customer.

See also

AddProjectDialog

8.65.3.4 void Gui::MainWindow::addQuote ( ) [slot]

MainWindow::addQuote open window to add a new quote.

See also

AddQuoteDialog

8.65.3.5 void Gui::MainWindow::editUser ( ) [slot]

MainWindow::editUser modify the user.

See also

UserDataDialog

8.65.3.6 int Gui::MainWindow::getCurrentCustomerId ( )

MainWindow::getCurrentCustomerId get the selected customer.

Returns

id of the selected customer

8.65.3.7 QString Gui::MainWindow::getCurrentCustomerName ( )

MainWindow::getCurrentCustomerName get the selected customer name in the customers' table.

Returns

name of the selected customer

8.65.3.8 int Gui::MainWindow::getCurrentProjectId ( )

MainWindow::getCurrentProjectId get the selected project id.

Returns

id of the selected project

8.65.3.9 QString Gui::MainWindow::getCurrentProjectName ( )

MainWindow::getCurrentProjectName get the selected project name in the table of projects.

Returns

name of the selected project

8.65.3.10 int Gui::MainWindow::getCurrentQuoteId ( )

MainWindow::getCurrentQuoteId get the selected quote id.

Returns

id of the selected quote

8.65.3.11 void Gui::MainWindow::resizeEvent (QResizeEvent * *event*)

MainWindow::resizeEvent Resize central TableView when you resize the **MainWindow**

Parameters

event	Resize event

8.65.3.12 QModelIndex Gui::MainWindow::rootTree ( )

MainWindow::rootTree return the root of the tree "Tous les clients".

Returns

QModelIndex

8.65.3.13 void Gui::MainWindow::search (QString *s*) [slot]

MainWindow::search launch a new search.

Parameters

s	text in field

8.65.3.14 int Gui::MainWindow::treeLevel ( )

MainWindow::treeLevel return the level of the node selected in the tree.

Returns

integer, depth of the item in tree

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

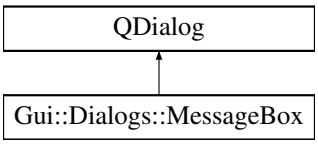
- src/gui/mainwindow/mainwindow.h
- src/gui/mainwindow/mainwindow.cpp

8.66 Gui::Dialogs::MessageBox Class Reference

The MessageBox class Information window with message.

`#include <messagebox.h>`

Inheritance diagram for Gui::Dialogs::MessageBox:



Public Member Functions

- MessageBox (QWidget *parent=0)

  *MessageBox::MessageBox Construt a **MessageBox***
- void aboutFact ()

  MessageBox::aboutFact Defines FACT team information.
- void aboutFactDev ()

  MessageBox::aboutFactDev Defines FactDev software information.
- void aboutIcons ()

  MessageBox::aboutIcons Defines icons theme information.
- void setImage (QString img, int width=128, int height=128)

  MessageBox::setImage Add the icon img to the current window.
- void setText (QString txt)

  MessageBox::setText Add the text txt to the current window.

Static Public Member Functions

- static void showAboutFact ()

  MessageBox::showAboutFact Shows window about FACT team.
- static void showAboutFactDev ()

  MessageBox::showAboutFactDev Shows window about FactDev software.
- static void showAboutIcons ()

  MessageBox::showAboutIcons Shows about icons theme of FactDev software.

8.66.1 Detailed Description

The MessageBox class Information window with message.

Author

Florent Berbie

8.66.2 Constructor & Destructor Documentation

8.66.2.1 Gui::Dialogs::MessageBox::MessageBox (QWidget * *parent* = `0`) `[explicit]`

MessageBox::MessageBox Construt a **MessageBox**

Parameters

parent	

8.66.3 Member Function Documentation

8.66.3.1 void Gui::Dialogs::MessageBox::setImage (QString *img*, int *width* = 128, int *height* = 128)

MessageBox::setImage Add the icon *img* to the current window.

Parameters

img	Icon
width	Icon width (default: 128)
height	Icon height (default: 128)

8.66.3.2 void Gui::Dialogs::MessageBox::setText (QString *txt*)

MessageBox::setText Add the text *txt* to the current window.

Parameters

txt	Text inside the current window

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

- src/gui/dialogs/messagebox.h
- src/gui/dialogs/messagebox.cpp

8.67 Parameters Class Reference

The Parameters class Class for simple user parameters.

`#include <parameters.h>`

Static Public Attributes

- static const QString DB_FILENAME = "database.db"
  DB_FILENAME The database sqlite file name.
- static const double VERSION = 1.0
  VERSION Version number of software.

8.67.1 Detailed Description

The Parameters class Class for simple user parameters.

Author

Antoine de Roquemaurel

8.67.2 Member Data Documentation

8.67.2.1 const QString Parameters::DB_FILENAME = "database.db" [static]

DB_FILENAME The database sqlite file name.

Database file name

8.67.2.2 const double Parameters::VERSION = 1.0 [static]

VERSION Version number of software.

Application version

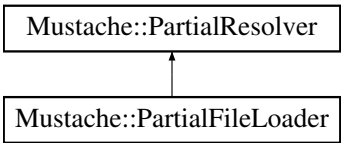
The documentation for this class was generated from the following files:

- src/parameters.h
- src/parameters.cpp

8.68 Mustache::PartialFileLoader Class Reference

`#include <mustache.h>`

Inheritance diagram for Mustache::PartialFileLoader:



Public Member Functions

- **PartialFileLoader** (const QString &basePath)
- virtual QString getPartial (const QString &name)

8.68.1 Detailed Description

A partial fetcher when loads templates from '<name>.mustache' files in a given directory.

Once a partial has been loaded, it is cached for future use.

8.68.2 Member Function Documentation

8.68.2.1 QString PartialFileLoader::getPartial (const QString & *name*) [virtual]

Returns the partial template with a given `name`.

Implements Mustache::PartialResolver.

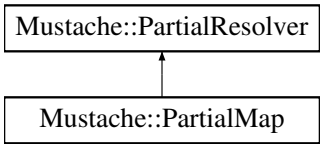
The documentation for this class was generated from the following files:

- src/libs/qt-mustache/src/mustache.h
- src/libs/qt-mustache/src/mustache.cpp

8.69 Mustache::PartialMap Class Reference

`#include <mustache.h>`

Inheritance diagram for Mustache::PartialMap:



Public Member Functions

- **PartialMap** (const QHash< QString, QString > &partials)
- virtual QString getPartial (const QString &name)

8.69.1 Detailed Description

A simple partial fetcher which returns templates from a map of (partial name -> template)

8.69.2 Member Function Documentation

8.69.2.1 QString PartialMap::getPartial (const QString & *name*) [virtual]

Returns the partial template with a given `name`.

Implements Mustache::PartialResolver.

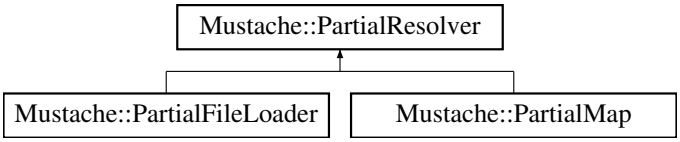
The documentation for this class was generated from the following files:

- src/libs/qt-mustache/src/mustache.h
- src/libs/qt-mustache/src/mustache.cpp

8.70 Mustache::PartialResolver Class Reference

`#include <mustache.h>`

Inheritance diagram for Mustache::PartialResolver:



Public Member Functions

- virtual QString getPartial (const QString &name)=0

8.70.1 Detailed Description

Interface for fetching template partials.

8.70.2 Member Function Documentation

8.70.2.1 virtual QString Mustache::PartialResolver::getPartial (const QString & *name*) [pure virtual]

Returns the partial template with a given `name`.

Implemented in Mustache::PartialFileLoader, and Mustache::PartialMap.

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

- src/libs/qt-mustache/src/mustache.h

8.71 Generator::PdfGenerator Class Reference

The PdfGenerator class Generator of PDF files.

`#include <pdfgenerator.h>`

Public Member Functions

- PdfGenerator (QString pdflatexPath="pdflatex")

  PdfGenerator::PdfGenerator Construct a PdfGenerator.
- void generate (QString inputDir, QString filename)

  PdfGenerator::generate Generate a PDF of the file named filename into the directory inputDir

8.71.1 Detailed Description

The PdfGenerator class Generator of PDF files.

8.71.2 Constructor & Destructor Documentation

8.71.2.1 Generator::PdfGenerator::PdfGenerator (QString *pdflatexPath* = `"pdflatex"`)

PdfGenerator::PdfGenerator Construct a PdfGenerator.

Parameters

pdflatexPath	Path to the command to generate PDF file

8.71.3 Member Function Documentation

8.71.3.1 void Generator::PdfGenerator::generate (QString *inputDir*, QString *filename*)

PdfGenerator::generate Generate a PDF of the file named *filename* into the directory *inputDir*

Parameters

inputDir	Directory where is store the PDF generated
filename	File name

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

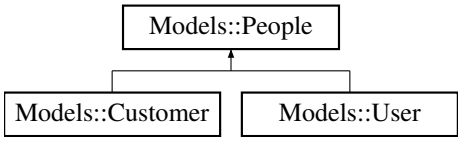
- src/generator/pdfgenerator.h
- src/generator/pdfgenerator.cpp

8.72 Models::People Class Reference

The People class People.

`#include <people.h>`

Inheritance diagram for Models::People:



Public Member Functions

- People ()

  People::People Construct a People.
- QString getFirstname () const

  People::getFirstname Return the People firstname.
- void setFirstname (const QString &firstname)

  People::setFirstnament Modify the People firstname
- QString getLastname () const

  People::getLastname Return the People lastname.
- void setLastname (const QString &lastname)

  People::setLastname Modify the People lastname
- QString getCompany () const

  People::getCompany Return the People company.
- void setCompany (const QString &company)

  People::setCompany Modify the People company name.
- QString getAddress () const

  People::getAddress Return the company addess (Number and name of street)
- void setAddress (const QString &address)

  People::setAddress Modify the People company address
- QString getPostalCode () const

  People::getPostalCode Return the postal code.
- void setPostalCode (const QString &postalCode)

  People::setPostalCode Modify the postal code postalCode
- QString getCity () const

  People::getCity Return the city.
- void setCity (const QString &city)

  People::setCity Modify the city
- QString getCountry () const

  People::getCountry Return the country of the People.
- void setCountry (const QString &country)

  People::setCountry Replace the country of the People by country
- QString getEmail () const

  People::getEmail Return the People professional email
- void setEmail (const QString &email)

  People::setEmail Modify the People professional email
- QString getPhone () const

  People::getPhone Return the number of the desktop phone.

- void setPhone (const QString &phone)

  People::setPhone Modify the number of the desktop phone
- QString getMobilePhone () const

  People::getMobilePhone Return the number of the professional mobile phone.
- void setMobilePhone (const QString &mobilePhone)

  People::setMobilePhone Modify the number of the professional People mobile phone mobilePhone
- QString getFax () const

  People::getFax Return the fax number.
- void setFax (const QString &fax)

  People::setFax Replace the current fax number by fax
- QString getAddressComplement () const

  getAddressComplement Return the address complement (Building, Appartment, . . .)
- void setAddressComplement (const QString &addressComplement)

  setAddressComplement Change the address complement
- QString getWebsite () const

  getWebsite Website of People
- void setWebsite (const QString &website)

  setWebsite Change the url of website
- bool operator== (const People &c)

  *People::operator == Re-define the operator "==" to compare if the current People is the same to the other **People** c Return TRUE if both Peoples are the same, else FALSE.*
- bool operator!= (const People &c)

  *People::operator == Re-define the operator "!=" to compare if the current People is differnt to the other **People** c Return TRUE if both Peoples are different, else FALSE.*

8.72.1 Detailed Description

The People class People.

8.72.2 Member Function Documentation

8.72.2.1 QString Models::People::getAddress ( ) const

People::getAddress Return the company addess (Number and name of street)

Returns

Address company

8.72.2.2 QString Models::People::getAddressComplement ( ) const

getAddressComplement Return the address complement (Building, Appartment, . . .)

Returns

The address complement

8.72.2.3 QString Models::People::getCity ( ) const

People::getCity Return the city.

Returns

city

8.72.2.4 QString Models::People::getCompany ( ) const

People::getCompany Return the People company.

Returns

New company name

8.72.2.5 QString Models::People::getCountry ( ) const

People::getCountry Return the country of the People.

Returns

country of the People

8.72.2.6 QString Models::People::getEmail ( ) const

People::getEmail Return the People professional *email*

Returns

professional email

8.72.2.7 QString Models::People::getFax ( ) const

People::getFax Return the fax number.

Returns

fax number

8.72.2.8 QString Models::People::getFirstname ( ) const

People::getFirstname Return the People firstname.

Returns

People firstname

8.72.2.9 QString Models::People::getLastname ( ) const

People::getLastname Return the People lastname.

Returns

People lastname

8.72.2.10 QString Models::People::getMobilePhone ( ) const

People::getMobilePhone Return the number of the professional mobile phone.

Returns

number of mobile phone

8.72.2.11 QString Models::People::getPhone ( ) const

People::getPhone Return the number of the desktop phone.

Returns

number of the desktop phone

8.72.2.12 QString Models::People::getPostalCode ( ) const

People::getPostalCode Return the postal code.

Returns

postal code

8.72.2.13 QString Models::People::getWebsite ( ) const

getWebsite Website of People

Returns

The website url

8.72.2.14 bool Models::People::operator!= (const People & *c*)

People::operator == Re-define the operator "!=" to compare if the current People is differnt to the other **People** *c* Return TRUE if both Peoples are different, else FALSE.

Parameters

c	People to compare

Returns

boolean

8.72.2.15 bool Models::People::operator== (const People & *c*)

People::operator == Re-define the operator "==" to compare if the current People is the same to the other **People** *c* Return TRUE if both Peoples are the same, else FALSE.

Parameters

c	People to compare

Returns

boolean

8.72.2.16 void Models::People::setAddress (const QString & *address*)

People::setAddress Modify the People company *address*

Parameters

address	Company address (name and number of street)

8.72.2.17 void Models::People::setAddressComplement (const QString & *addressComplement*)

setAddressComplement Change the address complement

Parameters

addressComplement	The new complement

8.72.2.18 void Models::People::setCity (const QString & *city*)

People::setCity Modify the *city*

Parameters

city	Company city address

8.72.2.19 void Models::People::setCompany (const QString & *company*)

People::setCompany Modify the People *company* name.

Parameters

company	New People company name

8.72.2.20 void Models::People::setCountry (const QString & *country*)

People::setCountry Replace the country of the People by *country*

Parameters

country	New country of the People

8.72.2.21 void Models::People::setEmail (const QString & *email*)

People::setEmail Modify the People professional *email*

Parameters

email	The People professional email

8.72.2.22 void Models::People::setFax (const QString & *fax*)

People::setFax Replace the current fax number by *fax*

Parameters

fax	new fax number

8.72.2.23 void Models::People::setFirstname (const QString & *firstname*)

People::setFirstnament Modify the People *firstname*

Parameters

firstname	New People firstname

8.72.2.24 void Models::People::setLastname (const QString & *lastname*)

People::setLastname Modify the People *lastname*

Parameters

lastname	New People lastname

8.72.2.25 void Models::People::setMobilePhone (const QString & *mobilePhone*)

People::setMobilePhone Modify the number of the professional People mobile phone *mobilePhone*

Parameters

mobilePhone	Number of the professional mobile phone

8.72.2.26 void Models::People::setPhone (const QString & *phone*)

People::setPhone Modify the number of the desktop *phone*

Parameters

phone	Number of the desktop phone

8.72.2.27 void Models::People::setPostalCode (const QString & *postalCode*)

People::setPostalCode Modify the postal code *postalCode*

Parameters

postalCode	New postal code

8.72.2.28 void Models::People::setWebsite (const QString & *website*)

setWebsite Change the url of website

Parameters

website	The new URL

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

- src/models/people.h
- src/models/people.cpp

8.73 Utils::pointers Class Reference

Static Public Member Functions

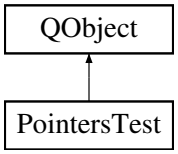
- static void **deleteIfNotNull** (QObject ∗p)

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

- src/utils/pointers.h
- src/utils/pointers.cpp

8.74 PointersTest Class Reference

Inheritance diagram for PointersTest:



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

- tests/utils/pointerstest.h
- tests/utils/pointerstest.cpp

8.75 Gui::Widgets::Popup Class Reference

Class for display popup quickly.

`#include <popup.h>`

Static Public Member Functions

- static void toImplement (QString, QWidget ∗)

  Popup::toImplement Method to display a critical message : feature is not implemented now.

8.75.1 Detailed Description

Class for display popup quickly.

Author

Antoine de Roquemaurel

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

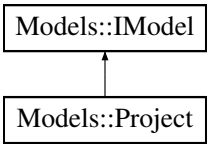
- src/gui/widgets/popup.h
- src/gui/widgets/popup.cpp

8.76 Models::Project Class Reference

The Project class : Project linked to a Customer.

`#include <project.h>`

Inheritance diagram for Models::Project:



Public Member Functions

- Project ()

  Project::Project Construct a Project.
- Project (QString name)

  Project::project Construct a project with a name.
- Project (int id)

  Project::Project Construct a Project which is specified by an id
- virtual ~Project ()

  ~Project Desctruct project object
- void commit ()

  Project::commit Update project data in the database.
- void hydrat (int id)

  Project::hydrat Insert project data which is specified by id in the database.
- void remove ()

  Project::remove Remove the current project.
- QVariantHash getDataMap ()

  getDataMap Get all data of model with a HashMap key/value
- QString getName () const

  Project::getName Return the project name.
- void setName (const QString &name)

  Project::setName Modify the project name
- QString getDescription () const

  Project::getDescription Return a project description.
- void setDescription (const QString &description)

  Project::setDescription Modify the project description
- QDate getBeginDate () const

  *Project::getBeginDate return the date of creation of the **Project***
- void setBeginDate (QDate beginDate)

  *Project::setBeginDate Modify beginDate of a **Project***
- QDate getEndDate () const

  *Project::getEndDate Return the endDate of the **Project***
- void setEndDate (QDate endDate)

  *Project::setEndDate Modify endDate of **Project***
- double getCost () const

  *Project::getCost Return the **Project** cost*
- void setCost (double cost)

  *Project::setCost Modify the **Project** cost*

- double getDailyRate () const

  Project::getDailyRate Return the daily rate estimated for this project.
- void setDailyRate (double dailyRate)

  Project::setDailyRate Modify the daily rate dailyRate of the current project.
- QSharedPointer< Customer > getCustomer () const

  Project::getCustomer Return the reference to the customer linked to this project.
- void setCustomer (QSharedPointer< Customer > customer)

  Project::setCustomer Modify the customer linked to this project.
- bool operator== (const Project &p)

  *Project::operator == Re-define the operator "==" to compare if the current project is the same to the other **Project** p Return TRUE if both projects are the same, else FALSE.*
- bool operator< (const Project &p) const

  *Project::operator < defines the operator "< to compare two **Project** and to see if the fisrt is anterior to the second.*
- bool operator!= (const Project &p)

  *Project::operator == Re-define the operator "!=" to compare if the current project is differnt to the other **Project** p Return TRUE if both projects are different, else FALSE.*
- double getCost ()

  *Project::costCompute compute the **Project** cost*

Additional Inherited Members

8.76.1 Detailed Description

The Project class : Project linked to a Customer.

Author

Florent Berbie

See also

IModel

8.76.2 Constructor & Destructor Documentation

8.76.2.1 Models::Project::Project (int *id*)

Project::Project Construct a Project which is specified by an *id*

Parameters

id	

8.76.3 Member Function Documentation

8.76.3.1 QDate Models::Project::getBeginDate ( ) const

Project::getBeginDate return the date of creation of the **Project**

Returns

the begin date of the Project

8.76.3.2 double Models::Project::getCost ( ) const

Project::getCost Return the **Project** *cost*

Returns

the project cost

8.76.3.3 double Models::Project::getCost ( )

Project::costCompute compute the **Project** *cost*

Returns

the project cost

8.76.3.4 QSharedPointer< Customer > Models::Project::getCustomer ( ) const

Project::getCustomer Return the reference to the customer linked to this project.

Returns

customer linked to this project

8.76.3.5 double Models::Project::getDailyRate ( ) const

Project::getDailyRate Return the daily rate estimated for this project.

Returns

the daily rate linket to the current project

8.76.3.6 QVariantHash Models::Project::getDataMap ( ) [virtual]

getDataMap Get all data of model with a HashMap key/value

Returns

Model's data

Implements Models::IModel.

8.76.3.7 QString Models::Project::getDescription ( ) const

Project::getDescription Return a project description.

Returns

project description

8.76.3.8 QDate Models::Project::getEndDate ( ) const

Project::getEndDate Return the *endDate* of the **Project**

Returns

the end date of the project

8.76.3.9 QString Models::Project::getName ( ) const

Project::getName Return the project name.

Returns

project name

8.76.3.10 void Models::Project::hydrat (int *id*) [virtual]

Project::hydrat Insert project data which is specified by *id* in the database.

Parameters

id	Project identify

Implements Models::IModel.

8.76.3.11 bool Models::Project::operator!= (const Project & *p*)

Project::operator == Re-define the operator "!=" to compare if the current project is differnt to the other **Project** *p* Return TRUE if both projects are different, else FALSE.

Parameters

c	Project to compare

Returns

boolean

8.76.3.12 bool Models::Project::operator< (const Project & *p*) const

Project::operator < defines the operator "< to compare two **Project** and to see if the fisrt is anterior to the second.

Parameters

b	the **Project** to compare with the current **Project**

Returns

true if the **Project** are different else false

8.76.3.13 bool Models::Project::operator== (const Project & *p*)

Project::operator == Re-define the operator "==" to compare if the current project is the same to the other **Project** *p* Return TRUE if both projects are the same, else FALSE.

Parameters

c	Project to compare

Returns

boolean

8.76.3.14 void Models::Project::setBeginDate (QDate *beginDate*)

Project::setBeginDate Modify *beginDate* of a **Project**

Parameters

beginDate	the new date of creation of the project

8.76.3.15 void Models::Project::setCost (double *cost*)

Project::setCost Modify the **Project** *cost*

Parameters

cost	the projectCost

8.76.3.16 void Models::Project::setCustomer (QSharedPointer< **Customer** > *customer*)

Project::setCustomer Modify the *customer* linked to this project.

Parameters

customer	New customer associated to this project

8.76.3.17 void Models::Project::setDailyRate (double *dailyRate*)

Project::setDailyRate Modify the daily rate *dailyRate* of the current project.

Parameters

dailyRate	New daily rate associated to the current project

8.76.3.18 void Models::Project::setDescription (const QString & *description*)

Project::setDescription Modify the project *description*

Parameters

description	New project description

8.76.3.19 void Models::Project::setEndDate (QDate *endDate*)

Project::setEndDate Modify *endDate* of **Project**

Parameters

endDate	the new end date of the project

8.76.3.20 void Models::Project::setName (const QString & *name*)

Project::setName Modify the project *name*

Parameters

name	Project name

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

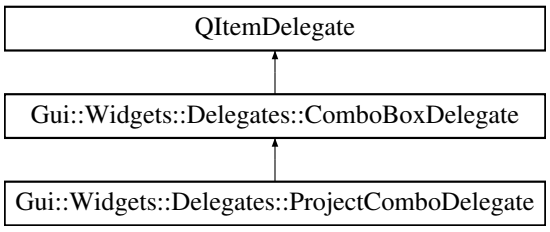
- src/models/project.h
- src/models/project.cpp

8.77 Gui::Widgets::Delegates::ProjectComboDelegate Class Reference

The ProjectComboDelegate class.

`#include <projectcombodelegate.h>`

Inheritance diagram for Gui::Widgets::Delegates::ProjectComboDelegate:



Public Member Functions

- ProjectComboDelegate (QSharedPointer< Models::Customer > c, QObject *parent=0)

  ProjectComboDelegate::ProjectComboDelegate.
- QWidget * createEditor (QWidget *parent, const QStyleOptionViewItem &option, const QModelIndex &index) const

  ProjectComboDelegate:createEditor Return a ComboBox specified by index item defined by the parent widget and style option which are used to control how the editor widgets appears.
- void paint (QPainter *painter, const QStyleOptionViewItem &option, const QModelIndex &index) const

  ProjectComboDelegate::paint Renders the delegate using the given painter and style option for the item specified by index
- void removeInCombo (QList< int > &l)

  ProjectComboDelegate::removeInCombo Remove the items contained in the list l into the current ComboBox.
- QMap< int, Models::Project > getProjects () const

  ProjectComboDelegate::getProjects Return a list of projects linked to a Customer ID.
- bool isLocked () const

  ProjectComboDelegate::isLocked Return TRUE if is locked else FALSE.
- void setLocked (bool locked)

  ProjectComboDelegate::setLocked Change the state of the lock by a new value locked

8.77.1 Detailed Description

The ProjectComboDelegate class.

8.77.2 Constructor & Destructor Documentation

8.77.2.1 Gui::Widgets::Delegates::ProjectComboDelegate::ProjectComboDelegate (QSharedPointer< Models::Customer > *c*, QObject * *parent* = 0)

ProjectComboDelegate::ProjectComboDelegate.

Parameters

c	
parent	

8.77.3 Member Function Documentation

8.77.3.1 QWidget * Gui::Widgets::Delegates::ProjectComboDelegate::createEditor (QWidget * *parent*, const QStyleOptionViewItem & *option*, const QModelIndex & *index*) const [virtual]

ProjectComboDelegate:createEditor Return a ComboBox specified by *index* item defined by the *parent* widget and style *option* which are used to control how the editor widgets appears.

Parameters

parent	Widget parent
option	Option style
index	Index for editing

Returns

ComboBox

Implements Gui::Widgets::Delegates::ComboBoxDelegate.

8.77.3.2 QMap< int, Models::Project > Gui::Widgets::Delegates::ProjectComboDelegate::getProjects ( ) const

ProjectComboDelegate::getProjects Return a list of projects linked to a Customer ID.

Returns

Projets of a Customer

8.77.3.3 bool Gui::Widgets::Delegates::ProjectComboDelegate::isLocked ( ) const

ProjectComboDelegate::isLocked Return TRUE if is locked else FALSE.

Returns

boolean

8.77.3.4 void Gui::Widgets::Delegates::ProjectComboDelegate::paint (QPainter * *painter*, const QStyleOptionViewItem & *option*, const QModelIndex & *index*) const [virtual]

ProjectComboDelegate::paint Renders the delegate using the given *painter* and style *option* for the item specified by *index*

Parameters

parent	Widget parent
option	Option style
index	Index for editing

Implements Gui::Widgets::Delegates::ComboBoxDelegate.

8.77.3.5 void Gui::Widgets::Delegates::ProjectComboDelegate::setLocked (bool *locked*)

ProjectComboDelegate::setLocked Change the state of the lock by a new value *locked*

Parameters

locked	

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

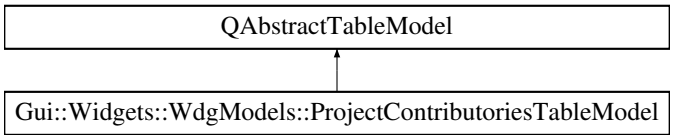
- src/gui/widgets/delegates/projectcombodelegate.h
- src/gui/widgets/delegates/projectcombodelegate.cpp

8.78 Gui::Widgets::WdgModels::ProjectContributoriesTableModel Class Reference

The ProjectContributoriesTableModel class Table model of contributories linked to projets.

`#include <projectcontributoriestablemodel.h>`

Inheritance diagram for Gui::Widgets::WdgModels::ProjectContributoriesTableModel:



Public Member Functions

- ProjectContributoriesTableModel (QObject *parent=0)

  ProjectContributoriesTableModel::ProjectContributoriesTableModel Construct a ProjectContributoriesTableModel.
- int rowCount (const QModelIndex &) const

  ProjectContributoriesTableModel::rowCount Number of contributories row.
- int columnCount (const QModelIndex &) const

  ProjectContributoriesTableModel::columnCount Number of column of a contributory.
- QVariant data (const QModelIndex &index, int role) const

  ProjectContributoriesTableModel::data Obtains data of a specify cell.
- QVariant headerData (int section, Qt::Orientation orientation, int role) const

  ProjectContributoriesTableModel::headerData Obtains header title of table.
- bool setData (const QModelIndex &index, const QVariant &value, int role=Qt::EditRole)

  ProjectContributoriesTableModel::setData Change data of a cell.
- Qt::ItemFlags flags (const QModelIndex &index) const

  ProjectContributoriesTableModel::flags Differents table flags.
- void append (QPair< Models::Project *, Models::Rate > p)

  ProjectContributoriesTableModel::append Add a new rate p to a Project.
- void append ()

ProjectContributoriesTableModel::append *Add the current element to the list.*

- bool allProjectsChosen ()

  ProjectContributoriesTableModel::allProjectsChosen *Return TRUE if all projects have been selected else FALSE.*

- QList< int > & getSelectedProjects ()

  ProjectContributoriesTableModel::getSelectedProjects *Return the list of the Project id which are selected.*

- void remove (int index)

  ProjectContributoriesTableModel::remove *Remove the element to the index*

- QPair< Models::Project *, Models::Rate > getProject (const int row)

  ProjectContributoriesTableModel::getProject *Return the project and it rate linked which correspond to the row*

- QList< QPair< Models::Project *, Models::Rate > > getProjects (void)

  ProjectContributoriesTableModel::getProjects *Return a list of the set of projects and their rates linked.*

8.78.1 Detailed Description

The ProjectContributoriesTableModel class Table model of contributories linked to projets.

Author

Antoine de Roquemaurel

8.78.2 Constructor & Destructor Documentation

8.78.2.1 Gui::Widgets::WdgModels::ProjectContributoriesTableModel::ProjectContributoriesTableModel (QObject * *parent* = 0)

ProjectContributoriesTableModel::ProjectContributoriesTableModel Construct a ProjectContributoriesTableModel.

Parameters

parent	Parent widget

8.78.3 Member Function Documentation

8.78.3.1 bool Gui::Widgets::WdgModels::ProjectContributoriesTableModel::allProjectsChosen ( )

ProjectContributoriesTableModel::allProjectsChosen Return TRUE if all projects have been selected else FALSE.

Returns

boolean All projects selected

8.78.3.2 void Gui::Widgets::WdgModels::ProjectContributoriesTableModel::append (QPair< Models::Project *, Models::Rate > *p*)

ProjectContributoriesTableModel::append Add a new rate *p* to a Project.

Parameters

p	

8.78.3.3 int Gui::Widgets::WdgModels::ProjectContributoriesTableModel::columnCount (const QModelIndex &) const

ProjectContributoriesTableModel::columnCount Number of column of a contributory.

Returns

The number of column

8.78.3.4 QVariant Gui::Widgets::WdgModels::ProjectContributoriesTableModel::data (const QModelIndex & *index*, int *role*) const

ProjectContributoriesTableModel::data Obtains data of a specify cell.

Parameters

index	The cell who we want data
role	The role of set

Returns

The data of cell

8.78.3.5 Qt::ItemFlags Gui::Widgets::WdgModels::ProjectContributoriesTableModel::flags (const QModelIndex & *index*) const

ProjectContributoriesTableModel::flags Differents table flags.

Parameters

index	The cell who we want to know flags

Returns

Flags

8.78.3.6 QPair< Models::Project *, Models::Rate > Gui::Widgets::WdgModels::ProjectContributoriesTableModel::getProject (const int *row*)

ProjectContributoriesTableModel::getProject Return the project and it rate linked which correspond to the *row*

Parameters

row	Row of the project to get

Returns

a Project and it rate

8.78.3.7 QList< QPair< Models::Project *, Models::Rate > > Gui::Widgets::WdgModels::ProjectContributoriesTableModel::getProjects (void)

ProjectContributoriesTableModel::getProjects Return a list of the set of projects and their rates linked.

Returns

List of Projects and Rates linked

8.78.3.8 QList< int > & Gui::Widgets::WdgModels::ProjectContributoriesTableModel::getSelectedProjects ( )

ProjectContributoriesTableModel::getSelectedProjects Return the list of the Project id which are selected.

Returns

List of ID Project

8.78.3.9 QVariant Gui::Widgets::WdgModels::ProjectContributoriesTableModel::headerData (int *section*, Qt::Orientation *orientation*, int *role*) const

ProjectContributoriesTableModel::headerData Obtains header title of table.

Parameters

section	The number of column
orientation	The table orientation
role	

Returns

The Title header of column

8.78.3.10 void Gui::Widgets::WdgModels::ProjectContributoriesTableModel::remove (int *index*)

ProjectContributoriesTableModel::remove Remove the element to the *index*

Parameters

index	Index of the element to remove

8.78.3.11 int Gui::Widgets::WdgModels::ProjectContributoriesTableModel::rowCount (const QModelIndex &) const

ProjectContributoriesTableModel::rowCount Number of contributories row.

Returns

The number of contributories

8.78.3.12 bool Gui::Widgets::WdgModels::ProjectContributoriesTableModel::setData (const QModelIndex & *index*, const QVariant & *value*, int *role* = `Qt::EditRole`)

ProjectContributoriesTableModel::setData Change data of a cell.

Parameters

index	The cell to change data
value	The new value
role	THe role of cell

Returns

True if we could edit

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

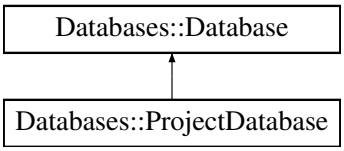
- src/gui/widgets/widgetsmodels/projectcontributoriestablemodel.h
- src/gui/widgets/widgetsmodels/projectcontributoriestablemodel.cpp

8.79 Databases::ProjectDatabase Class Reference

The ProjectDatabase class Project table database.

`#include <projectdatabase.h>`

Inheritance diagram for Databases::ProjectDatabase:



Public Member Functions

- Models::Project * getProject (const int pId)

  ProjectDatabase::getProject Get informations about the project identified by 'pId'.
- Models::Project * getProject (QSqlQuery &q)

  ProjectDatabase::getProject.
- int addProject (const Models::Project &)

  ProjectDatabase:addProject Add the project 'pProject' to the database.
- void updateProject (const Models::Project &)

  ProjectDatabase:updateProject Update informations about the project.
- void removeProject (const int pId)

  removeProject Remove the project with the id 'pId'
- int getNbProjects ()

  ProjectDatabase:getNbProjects Return the number of projects existing.
- int getNbProjectsForACustomer (const int pId)

  ProjectDatabase:getNbProjectsForACustomer Return the number of projects existing for an identify customer pId
- QMap< int, Models::Project > getProjectsOfCustomer (QSharedPointer< Models::Customer > c)

  getProjectsOfCustomer Return all projects of a customer
- QList< Models::Project > getProjects (const int customerId)

  getProjects Return all projects of a customer
- WdgModels::ProjectsTableModel * getProjectsTable (const int pId) throw (DbException*)

  getProjectsTable Return all projects of a customer in QStandardItemModel
- QSharedPointer< Project > updateProject (QSqlQuery &q)

  getProject Obtain a project without new query
- QList< Project * > getAllProjects ()

  Project::getAllProjects Return the list of projects.
- double getCostProjects (QList< Project * > projects)

  ProjectDatabase::getCostProjects compute the cost of the project list given in parameter.

Static Public Member Functions

- static ProjectDatabase * instance () throw (DbException*)

  ProjectDatabase::getInstance Return an instance of ProjectDatabase.

Additional Inherited Members

8.79.1 Detailed Description

The ProjectDatabase class Project table database.

Author

Florent Berbie

See also

Database
Project

8.79.2 Member Function Documentation

8.79.2.1 int Databases::ProjectDatabase::addProject (const Models::Project & *pProject*)

ProjectDatabase:addProject Add the project 'pProject' to the database.

Returns

project id

8.79.2.2 QList< Project * > Databases::ProjectDatabase::getAllProjects ( )

Project::getAllProjects Return the list of projects.

Returns

Projects list

8.79.2.3 double Databases::ProjectDatabase::getCostProjects (QList< Project * > *projects*)

ProjectDatabase::getCostProjects compute the cost of the project list given in parameter.

Parameters

projects	the list of projects

Returns

the cost of the list given

8.79.2.4 int Databases::ProjectDatabase::getNbProjects ( )

ProjectDatabase:getNbProjects Return the number of projects existing.

Returns

number of projects

8.79.2.5 int Databases::ProjectDatabase::getNbProjectsForACustomer (const int *pId*)

ProjectDatabase:getNbProjectsForACustomer Return the number of projects existing for an identify customer *pId*

Parameters

pId	Project id

Returns

number of projects

8.79.2.6 Models::Project * Databases::ProjectDatabase::getProject (const int *pId*)

ProjectDatabase::getProject Get informations about the project identified by 'pId'.

Parameters

pId	project

Returns

the project

8.79.2.7 Models::Project * Databases::ProjectDatabase::getProject (QSqlQuery & *q*)

ProjectDatabase::getProject.

Parameters

q	

Returns

8.79.2.8 QList< Project > Databases::ProjectDatabase::getProjects (const int *customerId*)

getProjects Return all projects of a customer

Parameters

customerId	The customer id

Returns

List with all projects of customer

8.79.2.9 QMap< int, Models::Project > Databases::ProjectDatabase::getProjectsOfCustomer (QSharedPointer< Models::Customer > *c*)

getProjectsOfCustomer Return all projects of a customer

Parameters

c	The customer

Returns

All projects of c with id in key

8.79.2.10 WdgModels::ProjectsTableModel * Databases::ProjectDatabase::getProjectsTable (const int *pId*) throw DbException *)

getProjectsTable Return all projects of a customer in QStandardItemModel

Parameters

filter	Select only projects who are specified by *filter*

Returns

QStandardItemModel an item model for QTableView

8.79.2.11 ProjectDatabase * Databases::ProjectDatabase::instance ( ) throw DbException *) [static]

ProjectDatabase::getInstance Return an instance of ProjectDatabase.

Returns

Instance of ProjectDatabase

8.79.2.12 void Databases::ProjectDatabase::removeProject (const int *pId*)

removeProject Remove the project with the id 'pId'

Parameters

pId	project id

8.79.2.13 QSharedPointer< Models::Project > Databases::ProjectDatabase::updateProject (QSqlQuery & *q*)

getProject Obtain a project without new query

Parameters

q	The query to use

Returns

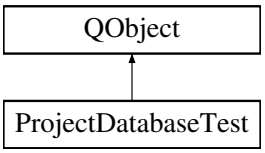
The project linked to q

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

- src/database/projectdatabase.h
- src/database/projectdatabase.cpp

8.80 ProjectDatabaseTest Class Reference

Inheritance diagram for ProjectDatabaseTest:

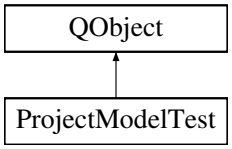


The documentation for this class was generated from the following files:

- tests/database/projectdatabasetest.h
- tests/database/projectdatabasetest.cpp

8.81 ProjectModelTest Class Reference

Inheritance diagram for ProjectModelTest:



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

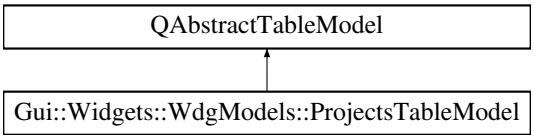
- tests/models/projectmodeltest.h
- tests/models/projectmodeltest.cpp

8.82 Gui::Widgets::WdgModels::ProjectsTableModel Class Reference

The ProjectsTableModel class for a Project table.

`#include <projectstablemodel.h>`

Inheritance diagram for Gui::Widgets::WdgModels::ProjectsTableModel:



Public Member Functions

- ProjectsTableModel ()

  ProjectsTableModel::ProjectsTableModel Construct a ProjectsTableModel.
- int rowCount (const QModelIndex &) const

  ProjectsTableModel::rowCount Number of projects row.
- int columnCount (const QModelIndex &) const

  ProjectsTableModel::columnCount Number of column of a Project.
- QVariant data (const QModelIndex &index, int role=Qt::DisplayRole) const

  ProjectsTableModel::data Obtains data of a specify cell.
- QVariant headerData (int section, Qt::Orientation orientation, int role=Qt::DisplayRole) const

  ProjectsTableModel::headerData Obtains header title of table.
- bool setData (const QModelIndex &index, const QVariant &value, int role=Qt::EditRole)

  ProjectsTableModel::setData Change data of a cell.
- void append (const Project &project)

  ProjectsTableModel::append Add a new line in table.
- void remove (const int i)

  ProjectsTableModel::remove Remove a line.
- Qt::ItemFlags flags (const QModelIndex &index) const

  ProjectsTableModel::flags Differents table flags.
- int count ()

  ProjectsTableModel::count Number of projects in table.
- QList< Project > getProjects () const

  ProjectsTableModel::getProjects Return the list of projects.

8.82.1 Detailed Description

The ProjectsTableModel class for a Project table.

Author

Florent Berbie

See also

Project

8.82.2 Constructor & Destructor Documentation

8.82.2.1 Gui::Widgets::WdgModels::ProjectsTableModel::ProjectsTableModel ( )

ProjectsTableModel::ProjectsTableModel Construct a ProjectsTableModel.

Parameters

parent	Parent widget

8.82.3 Member Function Documentation

8.82.3.1 void Gui::Widgets::WdgModels::ProjectsTableModel::append (const Project & *project*)

ProjectsTableModel::append Add a new line in table.

Parameters

Project	The new Project

8.82.3.2 int Gui::Widgets::WdgModels::ProjectsTableModel::columnCount (const QModelIndex &) const

ProjectsTableModel::columnCount Number of column of a Project.

Returns

The number of column

8.82.3.3 int Gui::Widgets::WdgModels::ProjectsTableModel::count ( )

ProjectsTableModel::count Number of projects in table.

Returns

The number of projects

8.82.3.4 QVariant Gui::Widgets::WdgModels::ProjectsTableModel::data (const QModelIndex & *index*, int *role* = `Qt::DisplayRole`) const

ProjectsTableModel::data Obtains data of a specify cell.

Parameters

index	The cell who we want data
role	The role of set

Returns

The data of cell

8.82.3.5 Qt::ItemFlags Gui::Widgets::WdgModels::ProjectsTableModel::flags (const QModelIndex & *index*) const

ProjectsTableModel::flags Differents table flags.

Parameters

index	The cell who we want to know flags

Returns

Flags

8.82.3.6 QList< Project > Gui::Widgets::WdgModels::ProjectsTableModel::getProjects (void) const

ProjectsTableModel::getProjects Return the list of projects.

Returns

list of projects

8.82.3.7 QVariant Gui::Widgets::WdgModels::ProjectsTableModel::headerData (int *section*, Qt::Orientation *orientation*, int *role* = `Qt::DisplayRole`) const

ProjectsTableModel::headerData Obtains header title of table.

Parameters

section	The number of column
orientation	The table orientation
role	

Returns

The Title header of column

8.82.3.8 void Gui::Widgets::WdgModels::ProjectsTableModel::remove (const int *i*)

ProjectsTableModel::remove Remove a line.

Parameters

i	The number of line to remove

8.82.3.9 int Gui::Widgets::WdgModels::ProjectsTableModel::rowCount (const QModelIndex &) const

ProjectsTableModel::rowCount Number of projects row.

Returns

The number of projects

8.82.3.10 bool Gui::Widgets::WdgModels::ProjectsTableModel::setData (const QModelIndex & *index*, const QVariant & *value*, int *role* = `Qt::EditRole`)

ProjectsTableModel::setData Change data of a cell.

Parameters

index	The cell to change data
value	The new value
role	The role of cell

Returns

True if we could edit

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

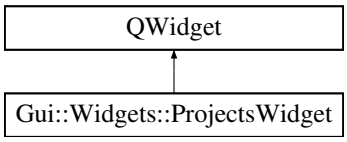
- src/gui/widgets/widgetsmodels/projectstablemodel.h
- src/gui/widgets/widgetsmodels/projectstablemodel.cpp

8.83 Gui::Widgets::ProjectsWidget Class Reference

The ProjectsWidget class Actions on Project.

`#include <projectswidget.h>`

Inheritance diagram for Gui::Widgets::ProjectsWidget:



Public Slots

- void newProject ()

  ProjectsWidget::addProject Event which sends a signal to add a new project.
- void editSelectedProject ()

  ProjectsWidget::editSelectedProject Event which sends a signal to edit the project selected.
- void removeSelectedProject ()

  ProjectsWidget::removeSelectedProject Event which sends a signal to remove the project selected.
- void updateBtn (bool b)

  ProjectsWidget::updateBtn Update the toolbar in tblProjects.

Signals

- void addProject ()

  ProjectsWidget::addProject Add a new project to the current Customer.
- void editProject ()

  ProjectsWidget::editProject Edit the current Customer selected.
- void removeProject ()

  ProjectsWidget::removeProject Remove the current Customer selected.

Public Member Functions

- ProjectsWidget (QWidget ∗parent=0)

  ProjectsWidget::ProjectsWidget Construct a ProjectsWidget.

8.83.1 Detailed Description

The ProjectsWidget class Actions on Project.

Author

Florent Berbie

8.83.2 Constructor & Destructor Documentation

8.83.2.1 Gui::Widgets::ProjectsWidget::ProjectsWidget (QWidget ∗ *parent* = 0) [explicit]

ProjectsWidget::ProjectsWidget Construct a ProjectsWidget.

Parameters

parent	

8.83.3 Member Function Documentation

8.83.3.1 void Gui::Widgets::ProjectsWidget::updateBtn (bool *b*) [slot]

ProjectsWidget::updateBtn Update the toolbar in tblProjects.

Parameters

boolean	if a row is selected

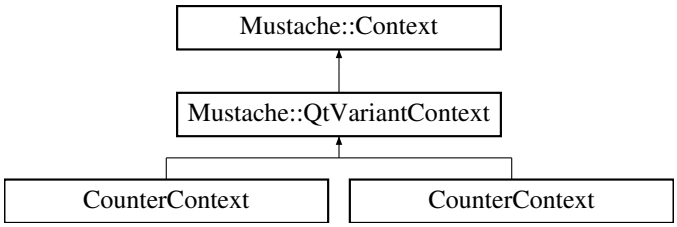
The documentation for this class was generated from the following files:

- src/gui/widgets/projectswidget.h
- src/gui/widgets/projectswidget.cpp

8.84 Mustache::QtVariantContext Class Reference

`#include <mustache.h>`

Inheritance diagram for Mustache::QtVariantContext:



Public Types

- typedef QString(∗ fn_t) (const QString &, Mustache::Renderer ∗, Mustache::Context ∗)

Public Member Functions

- **QtVariantContext** (const QVariant &root, PartialResolver ∗resolver=0)
- virtual QString stringValue (const QString &key) const
- virtual bool isFalse (const QString &key) const
- virtual int listCount (const QString &key) const
- virtual void push (const QString &key, int index=-1)
- virtual void pop ()
- virtual bool canEval (const QString &key) const
- virtual QString eval (const QString &key, const QString &_template, Mustache::Renderer ∗renderer)

8.84.1 Detailed Description

A context implementation which wraps a QVariantHash or QVariantMap.

8.84.2 Member Typedef Documentation

8.84.2.1 typedef QString(∗ Mustache::QtVariantContext::fn_t) (const QString &, Mustache::Renderer ∗, Mustache::Context ∗)

Construct a QtVariantContext which wraps a dictionary in a QVariantHash or a QVariantMap.

8.84.3 Member Function Documentation

8.84.3.1 bool QtVariantContext::canEval (const QString & *key*) const [virtual]

Returns true if eval() should be used to render section tags using `key`. If canEval() returns true for a key, the renderer will pass the literal, unrendered block of text for the section to eval() and replace the section with the result.

canEval() and eval() are equivalents for callable objects (eg. lambdas) in other Mustache implementations.

The default implementation always returns false.

Reimplemented from Mustache::Context.

Reimplemented in CounterContext, and CounterContext.

8.84.3.2 QString QtVariantContext::eval (const QString & *key,* const QString & *_template,* Mustache::Renderer ∗ *renderer*) [virtual]

Callback used to render a template section with the given `key`. `renderer` will substitute the original section tag with the result of eval().

The default implementation returns an empty string.

Reimplemented from Mustache::Context.

Reimplemented in CounterContext, and CounterContext.

8.84.3.3 bool QtVariantContext::isFalse (const QString & *key*) const [virtual]

Returns true if the value for `key` is 'false' or an empty list. 'False' values typically include empty strings, the boolean value false etc.

When processing a section Mustache tag, the section is not rendered if the key is false, or for an inverted section tag, the section is only rendered if the key is false.

Implements Mustache::Context.

8.84.3.4 int QtVariantContext::listCount (const QString & *key*) const [virtual]

Returns the number of items in the list value for `key` or 0 if the value for `key` is not a list.

Implements Mustache::Context.

8.84.3.5 void QtVariantContext::pop ( ) [virtual]

Exit the current context.

Implements Mustache::Context.

8.84.3.6 void QtVariantContext::push (const QString & *key*, int *index* = `-1`) [virtual]

Set the current context to the value for `key`. If index is >= 0, set the current context to the `index'th` value in the list value for `key`.

Implements Mustache::Context.

8.84.3.7 QString QtVariantContext::stringValue (const QString & *key*) const [virtual]

Returns a string representation of the value for `key` in the current context. This is used to replace a Mustache value tag.

Implements Mustache::Context.

Reimplemented in CounterContext, and CounterContext.

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

- src/libs/qt-mustache/src/mustache.h
- src/libs/qt-mustache/src/mustache.cpp

8.85 Models::Rate Class Reference

The Rate class Rate of a prestation.

`#include <rate.h>`

Public Member Functions

- Rate ()

Rate::Rate Construct a Rate.

- Rate (double hourly)

  Rate::Rate Construct a Rate with an hourly rate.
- Rate (int idBilling, int idProject)

  Rate::Rate Construct a Rate for the Billing with idBilling as identify and from the project identified by idProject
- void setDailyRate (const double &)

  Rate::setDailyRate Modify the daily rate with the new value.
- void setHourlyRate (const double &)

  Rate::setHourlyRate Modify the hourly rate with the new value.
- double getDailyRate () const

  Rate::getDailyRate Return the daily rate.
- double getHourlyRate () const

  Rate::getHourlyRate Return the hourly rate.
- int getNbDailyHours () const

  Rate::getNbDailyHours Return the number of daily hours.
- double getLegalRate () const

  Rate::getLegalRate Return the legal rate.

8.85.1 Detailed Description

The Rate class Rate of a prestation.

Author

Antoine de Roquemaurel

8.85.2 Constructor & Destructor Documentation

8.85.2.1 Models::Rate::Rate (double *hourly*)

Rate::Rate Construct a Rate with an hourly rate.

Parameters

hourly	Hourly rate

8.85.2.2 Models::Rate::Rate (int *idBilling*, int *idProject*)

Rate::Rate Construct a Rate for the Billing with *idBilling* as identify and from the project identified by *idProject*

Parameters

idBilling	Billing identify
idProject	Project identify

8.85.3 Member Function Documentation

8.85.3.1 double Models::Rate::getDailyRate ( ) const

Rate::getDailyRate Return the daily rate.

Returns

8.85.3.2 double Models::Rate::getHourlyRate ( ) const

Rate::getHourlyRate Return the hourly rate.

Returns

8.85.3.3 double Models::Rate::getLegalRate ( ) const

Rate::getLegalRate Return the legal rate.

Returns

8.85.3.4 int Models::Rate::getNbDailyHours ( ) const

Rate::getNbDailyHours Return the number of daily hours.

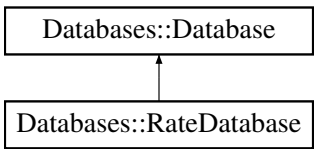
Returns

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

- src/models/rate.h
- src/models/rate.cpp

8.86 Databases::RateDatabase Class Reference

Inheritance diagram for Databases::RateDatabase:



Public Member Functions

- void addRateProject (int idProject, int idBilling, double hourlyRate)

  RateDatabase::addRateProject Add a new rate hourlyRate to the Project identified by idProject of the Billing with idBilling as identify.
- double getRate (const int idBilling, const int idProject) const

  RateDatabase::getRate Return the rate of the Project identified by idProject of the Billing with idBilling as identify.
- void updateRateProject (int idProject, int idBilling, double hourlyRate)

  RateDatabase::updateRateProject Update the current rate by the new hourlyRate of the Project identified by idProject of the Billing with idBilling as identify.

Static Public Member Functions

- static RateDatabase * instance () throw (DbException*)

  RateDatabase::getInstance Return an instance of RateDatabase.

Additional Inherited Members

8.86.1 Member Function Documentation

8.86.1.1 void Databases::RateDatabase::addRateProject (int *idProject*, int *idBilling*, double *hourlyRate*)

RateDatabase::addRateProject Add a new rate *hourlyRate* to the Project identified by *idProject* of the Billing with *idBilling* as identify.

Parameters

idProject	Project identify
idBilling	Billing identify
hourlyRate	New hourly rate

8.86.1.2 double Databases::RateDatabase::getRate (const int *idBilling*, const int *idProject*) const

RateDatabase::getRate Return the rate of the Project identified by *idProject* of the Billing with *idBilling* as identify.

Parameters

idBilling	Billing identify
idProject	Project identify

Returns

Rate of the Project of a Billing

8.86.1.3 RateDatabase * Databases::RateDatabase::instance ( ) throw DbException *) [static]

RateDatabase::getInstance Return an instance of RateDatabase.

Returns

Instance of RateDatabase

8.86.1.4 void Databases::RateDatabase::updateRateProject (int *idProject*, int *idBilling*, double *hourlyRate*)

RateDatabase::updateRateProject Update the current rate by the new *hourlyRate* of the Project identified by *id↩ Project* of the Billing with *idBilling* as identify.

Parameters

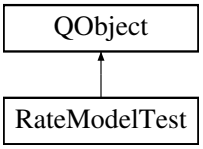
idProject	Project identify
idBilling	Billing identify
hourlyRate	New hourly rate

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

- src/database/ratedatabase.h
- src/database/ratedatabase.cpp

8.87 RateModelTest Class Reference

Inheritance diagram for RateModelTest:



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

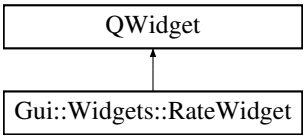
- tests/models/ratemodeltest.h
- tests/models/ratemodeltest.cpp

8.88 Gui::Widgets::RateWidget Class Reference

Class for display Rate.

`#include <ratewidget.h>`

Inheritance diagram for Gui::Widgets::RateWidget:



Public Slots

- void setDailyRate ()

  setDailyRate Set a new value for the daily rate
- void setHourlyRate ()

  setHourlyRate Set a new value for the hourly rate

Public Member Functions

- RateWidget (QWidget *parent=0)

  RateWidget::RateWidget Construct a rate widget.
- void initRate ()

  RateWidget::initRate Initialize the rate.
- void setWidgetDailyRateValue (double value)

  RateWidget::setWidgetDailyRateValue Modify the value of the daily rate spin box component.
- double getDailyRate ()

  RateWidget::getDailyRate Get the daily rate.
- double getHourlyRate ()

  RateWidget::getHourlyRate Get the hourly rate.
- void setWidgetHourlyRateValue (double value)

  RateWidget::setWidgetHourlyRateValue Modify the value of the hourly rate spin box component.
- void updateConversionRate ()

  updateConversionRate Update daily rate or hourly rate

8.88.1 Detailed Description

Class for display Rate.

Author

Florent Berbie

8.88.2 Constructor & Destructor Documentation

8.88.2.1 Gui::Widgets::RateWidget::RateWidget (QWidget * *parent* = 0) [explicit]

RateWidget::RateWidget Construct a rate widget.

Parameters

parent	The QWidget parent

8.88.3 Member Function Documentation

8.88.3.1 double Gui::Widgets::RateWidget::getDailyRate ( )

RateWidget::getDailyRate Get the daily rate.

Returns

The daily rate

8.88.3.2 double Gui::Widgets::RateWidget::getHourlyRate ( )

RateWidget::getHourlyRate Get the hourly rate.

Returns

The hourly rate

8.88.3.3 void Gui::Widgets::RateWidget::setDailyRate ( ) [slot]

setDailyRate Set a new value for the daily rate

Parameters

dailyRate	The new daily rate

8.88.3.4 void Gui::Widgets::RateWidget::setHourlyRate ( ) [slot]

setHourlyRate Set a new value for the hourly rate

Parameters

hourlyRate	The new hourly rate

8.88.3.5 void Gui::Widgets::RateWidget::setWidgetDailyRateValue (double *value*)

RateWidget::setWidgetDailyRateValue Modify the *value* of the daily rate spin box component.

Parameters

value	New Value

8.88.3.6 void Gui::Widgets::RateWidget::setWidgetHourlyRateValue (double *value*)

RateWidget::setWidgetHourlyRateValue Modify the *value* of the hourly rate spin box component.

Parameters

value	New value

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

- src/gui/widgets/ratewidget.h
- src/gui/widgets/ratewidget.cpp

8.89 Mustache::Renderer Class Reference

`#include <mustache.h>`

Public Member Functions

- QString render (const QString &_template, Context *context)
- QString error () const
- int errorPos () const
- QString errorPartial () const
- void setTagMarkers (const QString &startMarker, const QString &endMarker)

8.89.1 Detailed Description

Renders Mustache templates, replacing mustache tags with values from a provided context.

8.89.2 Member Function Documentation

8.89.2.1 QString Renderer::error ( ) const

Returns a message describing the last error encountered by the previous render() call.

8.89.2.2 QString Renderer::errorPartial ( ) const

Returns the name of the partial where the error occurred, or an empty string if the error occurred in the main template.

8.89.2.3 int Renderer::errorPos ( ) const

Returns the position in the template where the last error occurred when rendering the template or -1 if no error occurred.

If the error occurred in a partial template, the returned position is the offset in the partial template.

8.89.2.4 QString Renderer::render (const QString & *_template,* Context * *context*)

Render a Mustache template, using `context` to fetch the values used to replace Mustache tags.

8.89.2.5 void Renderer::setTagMarkers (const QString & *startMarker,* const QString & *endMarker*)

Sets the default tag start and end markers. This can be overridden within a template.

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

- src/libs/qt-mustache/src/mustache.h
- src/libs/qt-mustache/src/mustache.cpp

8.90 Models::Search Class Reference

The Search class.

`#include <search.h>`

Public Member Functions

- Search ()

  Search::Search Construct a search.
- ~Search ()

  Search::Search Destruct a search.
- QString getFilter ()

  Search::getFilter Return the search filter.
- void filterOnVarcharElements (QString &filter, const QStringList list, QString element)

  Search::filterOnVarcharElements Search, for each word of the list from the filter, if it corresponds to the element in the database.
- void filterOnNumberElements (QString &filter, const QStringList list, QString element)

  Search::filterOnNumberElements Search, for each number of the list from the filter, if it corresponds to the element in the database.
- void filterOnCompany (QString &filter, const QStringList list)

  Search::filterOnCompany Search all companies contained in the list of the filter
- void filterOnReferentLastname (QString &filter, const QStringList list)

  Search::filterOnReferentLastname Search all referents last name contained in the list of the filter
- void filterOnProjects (QString &filter, const QStringList list)

  Search::filterOnProjects Search all projects contained in the list of the filter
- void filterOnContributories (QString &filter, const QStringList list)

  Search::filterOnContributories Search all contributories contained in the list of the filter
- void filterOnBillsOrQuotes (QString &filter, const QStringList list)

  Search::filterOnBillsOrQuotes Search all bills or quotes which are contained in the list of the filter
- void filterOnCustomersWithoutProject (QString &filter, const QStringList list)

  Search::filterOnCustomersWithoutProject Search all customers which are contained in the list of the filter
- bool getSearchInCompanies () const

  Search::getSearchInCompanies Return if we search a company.
- void setSearchInCompanies (bool searchInCompanies)

  Search::setSearchInCompanies Modify the filter of companies search.
- bool getSearchInReferentLastname () const

  Search::getSearchInReferentLastname Return if we search a Last name referent.
- void setSearchInReferentLastname (bool searchInReferentLastname)

Search::setSearchInReferentLastname *Modify the filter of referents last name search.*

- bool getSearchInProjects () const

  Search::getSearchInProjects *Return TRUE if it exists one or more projets in database.*
- void setSearchInProjects (bool searchInProjects)

  Search::setSearchInProjects *Modify the filter of projects search.*
- bool searchInContributories () const

  Search::getSearchInContributories *Return TRUE if it exists one or more projets in database.*
- void setSearchInContributories (bool searchInContributories)

  Search::setSearchInContributories *Modify the filter of contributories search.*
- bool getSearchInBillsQuotes () const

  Search::searchInBillsQuotes *Return TRUE if it exists one or more bills or quotes in database.*
- void setSearchInBillsQuotes (bool searchInBillsQuotes)

  Search::setSearchInBillsQuotes *Modify the filter of bills and quotes search.*
- bool getGroupFilter () const

  Search::getGroupFilter *Return if the filter is actived.*
- void setGroupFilter (bool getGroupFilter)

  Search::setGroupFilter *Modify if we active search filter.*
- QString getText () const

  Search::getText *Return sql portion of filter.*
- void setText (const QString &getText)

  Search::setText *Modify sql portion.*

8.90.1 Detailed Description

The Search class.

Author

Antoine de Roquemaurel
Florent Berbie

8.90.2 Member Function Documentation

8.90.2.1 void Models::Search::filterOnBillsOrQuotes (QString & *filter,* const QStringList *list*)

Search::filterOnBillsOrQuotes Search all bills or quotes which are contained in the *list* of the *filter*

Parameters

filter	Text inputed in searchbar
list	List of bills or quotes

8.90.2.2 void Models::Search::filterOnCompany (QString & *filter,* const QStringList *list*)

Search::filterOnCompany Search all companies contained in the *list* of the *filter*

Parameters

filter	Text inputed in searchbar
list	List of companies

8.90.2.3 void Models::Search::filterOnContributories (QString & *filter,* const QStringList *list*)

Search::filterOnContributories Search all contributories contained in the *list* of the *filter*

Parameters

filter	Text inputed in searchbar
list	List of contributories

8.90.2.4 void Models::Search::filterOnCustomersWithoutProject (QString & *filter*, const QStringList *list*)

Search::filterOnCustomersWithoutProject Search all customers which are contained in the *list* of the *filter*

Parameters

filter	Text inputed in searchbar
list	List of bills or quotes

8.90.2.5 void Models::Search::filterOnNumberElements (QString & *filter*, const QStringList *list*, QString *element*)

Search::filterOnNumberElements Search, for each number of the *list* from the *filter*, if it corresponds to the *element* in the database.

Parameters

filter	Text inputed in the searchbar
list	List of numbers from the *filter*
element	Attribute name into the database

8.90.2.6 void Models::Search::filterOnProjects (QString & *filter*, const QStringList *list*)

Search::filterOnProjects Search all projects contained in the *list* of the *filter*

Parameters

filter	Text inputed in searchbar
list	List of projects

8.90.2.7 void Models::Search::filterOnReferentLastname (QString & *filter*, const QStringList *list*)

Search::filterOnReferentLastname Search all referents last name contained in the *list* of the *filter*

Parameters

filter	Text inputed in searchbar
list	List of referent last name

8.90.2.8 void Models::Search::filterOnVarcharElements (QString & *filter*, const QStringList *list*, QString *element*)

Search::filterOnVarcharElements Search, for each word of the *list* from the *filter*, if it corresponds to the *element* in the database.

Parameters

filter	Text inputed in the searchbar
list	List of words from the *filter*

element	Attribute name into the database

8.90.2.9 QString Models::Search::getFilter ( )

Search::getFilter Return the search filter.

Returns

filter selected (sql portion)

8.90.2.10 bool Models::Search::getGroupFilter ( ) const

Search::getGroupFilter Return if the filter is actived.

Returns

boolean if search filter is actived

8.90.2.11 bool Models::Search::getSearchInBillsQuotes ( ) const

Search::searchInBillsQuotes Return TRUE if it exists one or more bills or quotes in database.

Returns

boolean if bills or quotes are existing

8.90.2.12 bool Models::Search::getSearchInCompanies ( ) const

Search::getSearchInCompanies Return if we search a company.

Returns

boolean if we search a company

8.90.2.13 bool Models::Search::getSearchInProjects ( ) const

Search::getSearchInProjects Return TRUE if it exists one or more projets in database.

Returns

boolean if project are existing

8.90.2.14 bool Models::Search::getSearchInReferentLastname ( ) const

Search::getSearchInReferentLastname Return if we search a Last name referent.

Returns

boolean if search concerns the last name of referent

8.90.2.15 QString Models::Search::getText ( ) const

Search::getText Return sql portion of filter.

Returns

QString the sql portion

8.90.2.16 bool Models::Search::searchInContributories ( ) const

Search::getSearchInContributories Return TRUE if it exists one or more projets in database.

Returns

boolean if contributories are existing

8.90.2.17 void Models::Search::setGroupFilter (bool *getGroupFilter*)

Search::setGroupFilter Modify if we active search filter.

Parameters

getGroupFilter	Get if filter is actived

8.90.2.18 void Models::Search::setSearchInBillsQuotes (bool *searchInBillsQuotes*)

Search::setSearchInBillsQuotes Modify the filter of bills and quotes search.

Parameters

searchInBills↩ Quotes	Search in bills or quotes which are concerned

8.90.2.19 void Models::Search::setSearchInCompanies (bool *searchInCompanies*)

Search::setSearchInCompanies Modify the filter of companies search.

Parameters

getSearchIn↩ Companies	Search in companies is concerned

8.90.2.20 void Models::Search::setSearchInContributories (bool *searchInContributories*)

Search::setSearchInContributories Modify the filter of contributories search.

Parameters

searchIn↩ Contributories	Search in contributories which are concerned

8.90.2.21 void Models::Search::setSearchInProjects (bool *searchInProjects*)

Search::setSearchInProjects Modify the filter of projects search.

Parameters

searchInProjects	Search in projects which are concerned

8.90.2.22 void Models::Search::setSearchInReferentLastname (bool *searchInReferentLastname*)

Search::setSearchInReferentLastname Modify the filter of referents last name search.

Parameters

searchIn↩ Referent↩ Lastname	Search in referents last name which are concerned

8.90.2.23 void Models::Search::setText (const QString & *getText*)

Search::setText Modify sql portion.

Parameters

getText	Get sql portion

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

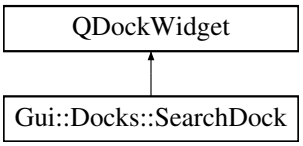
- src/models/search.h
- src/models/search.cpp

8.91 Gui::Docks::SearchDock Class Reference

The SearchDock class Dock which contains search bar.

`#include <searchdock.h>`

Inheritance diagram for Gui::Docks::SearchDock:



Public Slots

- void search (QString text)

  SearchDock::search Search elements in database which correspond to the text

Signals

- void textChanged (QString text)

  SearchDock::textChanged Current text changed by text

Public Member Functions

- SearchDock (QWidget ∗parent=0, Qt::WindowFlags flags=0)
  SearchDock::SearchDock Construct a SearchDock.

8.91.1 Detailed Description

The SearchDock class Dock which contains search bar.

8.91.2 Constructor & Destructor Documentation

8.91.2.1 Gui::Docks::SearchDock::SearchDock (QWidget ∗ *parent* = 0, Qt::WindowFlags *flags* = 0) [explicit]

SearchDock::SearchDock Construct a SearchDock.

Parameters

parent	Widget Parent
flags	Window flag

8.91.3 Member Function Documentation

8.91.3.1 void Gui::Docks::SearchDock::search (QString *text*) [slot]

SearchDock::search Search elements in database which correspond to the *text*

Parameters

text	Element to search

8.91.3.2 void Gui::Docks::SearchDock::textChanged (QString *text*) [signal]

SearchDock::textChanged Current text changed by *text*

Parameters

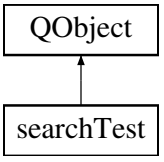
text	New text

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

- src/gui/docks/searchdock.h
- src/gui/docks/searchdock.cpp

8.92 searchTest Class Reference

Inheritance diagram for searchTest:



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

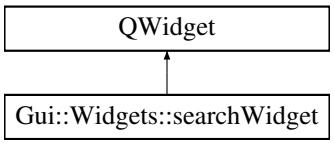
- tests/models/searchtest.h
- tests/models/searchtest.cpp

8.93 Gui::Widgets::searchWidget Class Reference

Class for search in database.

`#include <searchwidget.h>`

Inheritance diagram for Gui::Widgets::searchWidget:



Public Slots

- void search (QString toSearch)

  searchWidget::launch a search
- void getCustomerData ()

  searchWidget::getCustomerData Return data on the customer selected in the QTableView and display this data (Firstname, Lastname, Company)

Signals

- void selectCustomer ()

  searchWidget::selectCustomer Signal that the Customer selected has changed

Public Member Functions

- searchWidget (QWidget ∗parent=0)

  searchWidget::searchWidget Construct a search widget
- int getCurrentCustomerId ()

  searchWidget::getCurrentCustomerId Return the id of the customer selected in the table
- bool isCustomerSelected () const

  searchWidget::isCustomerSelected Return TRUE if a customer is selected, else FALSE
- void selectCustomer (int id)

  searchWidget::selectCustomer Select the Customer with it id
- int getIdCustomer () const

  searchWidget::getIdCustomer Return the Customer id of the Customer line selected
- void setIdCustomer (int idCustomer)

  searchWidget::setIdCustomer Change the customer id by the new idCustomer

8.93.1 Detailed Description

Class for search in database.

Author

Antoine de Roquemaurel

8.93.2 Constructor & Destructor Documentation

8.93.2.1 Gui::Widgets::searchWidget::searchWidget (QWidget * *parent* = 0) [explicit]

searchWidget::searchWidget Construct a search widget

Parameters

parent	The QWidget parent

8.93.3 Member Function Documentation

8.93.3.1 int Gui::Widgets::searchWidget::getCurrentCustomerId ( )

searchWidget::getCurrentCustomerId Return the id of the customer selected in the table

Returns

id of the current customer

8.93.3.2 int Gui::Widgets::searchWidget::getIdCustomer ( ) const

searchWidget::getIdCustomer Return the Customer id of the Customer line selected

Returns

The Customer ID

8.93.3.3 bool Gui::Widgets::searchWidget::isCustomerSelected ( ) const

searchWidget::isCustomerSelected Return TRUE if a customer is selected, else FALSE

Returns

boolean

8.93.3.4 void Gui::Widgets::searchWidget::search (QString *toSearch*) [slot]

searchWidget::launch a search

Parameters

toSearch	The value to search

8.93.3.5 void Gui::Widgets::searchWidget::selectCustomer (int *id*)

searchWidget::selectCustomer Select the Customer with it *id*

Parameters

id	Customer ID

8.93.3.6 void Gui::Widgets::searchWidget::setIdCustomer (int *idCustomer*)

searchWidget::setIdCustomer Change the customer id by the new *idCustomer*

Parameters

idCustomer	Customer id

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

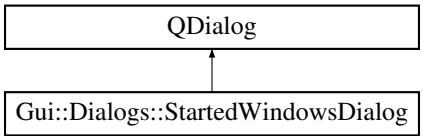
- src/gui/widgets/searchwidget.h
- src/gui/widgets/searchwidget.cpp

8.94 Gui::Dialogs::StartedWindowsDialog Class Reference

The StartedWindowsDialog class Contruct a Windows for the first begin.

`#include <startedwindowsdialog.h>`

Inheritance diagram for Gui::Dialogs::StartedWindowsDialog:



Public Slots

- void checkFields ()

  StartedWindowsDialog::checkFields Check all fields of dialog components.
- void backToPage2 ()

  StartedWindowsDialog::backToPage2 Return to the second page.
- void nextToPage2 ()

  StartedWindowsDialog::nextToPage2 Go to the second page.
- void nextToPage3 ()

  StartedWindowsDialog::nextToPage3 Go to the third page.
- void databaseTypeChanged (const int index)

  StartedWindowsDialog::databaseTypeChanged Listener on the Database type combobox.
- void databaseTypeChanged (void)

  StartedWindowsDialog::databaseTypeChanged Listener on the Database type combobox.
- void updateNextButton ()

  StartedWindowsDialog::updateNextButton Check if the next button is enabled.
- void accept ()

  StartedWindowsDialog::accept Valid data inputed by user and add these data in Database.

Public Member Functions

- StartedWindowsDialog (QWidget ∗parent=0)

  StartedWindowsDialog::StartedWindowsDialog Contruct a StartedWindowsDialog.
- void fillFields ()

  StartedWindowsDialog::fillFields Fill line edits with the data of the user.
- QPixmap getImage (QString path, int width=256, int height=256)

  StartedWindowsDialog::getImage Return a scaled image from the icon specified by it path. The image returned has a resolution of width∗height (default 256∗256)
- bool isDatabaseTypeValid ()

StartedWindowsDialog::isDatabaseTypeValid *Return TRUE if settings of database type is valid else return FALSE.*

- bool isDatabaseCentralized ()

  StartedWindowsDialog::isDatabaseCentralized *Return TRUE if the current type of database is centralized else FALSE.*

8.94.1 Detailed Description

The StartedWindowsDialog class Contruct a Windows for the first begin.

Author

Florent Berbie

See also

DatabaseSettingsWidget

8.94.2 Constructor & Destructor Documentation

8.94.2.1 Gui::Dialogs::StartedWindowsDialog::StartedWindowsDialog (QWidget * *parent* = 0) [explicit]

StartedWindowsDialog::StartedWindowsDialog Contruct a StartedWindowsDialog.

Parameters

parent	QWidget parent

8.94.3 Member Function Documentation

8.94.3.1 void Gui::Dialogs::StartedWindowsDialog::databaseTypeChanged (const int *index*) [slot]

StartedWindowsDialog::databaseTypeChanged Listener on the Database type combobox.

Parameters

index	Current index selected

8.94.3.2 QPixmap Gui::Dialogs::StartedWindowsDialog::getImage (QString *path*, int *width* = 256, int *height* = 256)

StartedWindowsDialog::getImage Return a scaled image from the icon specified by it *path*. The image returned has a resolution of *width*∗*height* (default 256∗256)

Parameters

path	Icon path
width	Icon width
height	Icon height

Returns

Scaled image

8.94.3.3 bool Gui::Dialogs::StartedWindowsDialog::isDatabaseCentralized ( )

StartedWindowsDialog::isDatabaseCentralized Return TRUE if the current type of database is centralized else F↩ ALSE.

Returns

boolean

8.94.3.4 bool Gui::Dialogs::StartedWindowsDialog::isDatabaseTypeValid ( )

StartedWindowsDialog::isDatabaseTypeValid Return TRUE if settings of database type is valid else return FALSE.

Returns

boolean on the database type validity

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

- src/gui/dialogs/startedwindowsdialog.h
- src/gui/dialogs/startedwindowsdialog.cpp

8.95 Models::Statistics Class Reference

Static Public Member Functions

- static QPair< int, double > getTurnoverBetweenDates (QDate begin, QDate end)
  getTurnoverBetweenDates Compute the turnover betweend 2 dates

8.95.1 Member Function Documentation

8.95.1.1 QPair< int, double > Statistics::getTurnoverBetweenDates (QDate *begin*, QDate *end*) [static]

getTurnoverBetweenDates Compute the turnover betweend 2 dates

Parameters

begin	The beginning date
end	The ending date

Returns

A Qpair who contains the number of billing and their costs

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

- src/models/statistics.h
- src/models/statistics.cpp

8.96 Utils::String Class Reference

The Utils class.

`#include <string.h>`

Static Public Member Functions

- static QString firstLetterToUpper (QString s)

  firstLetterToUpper Put the first letter of a string in capslock

8.96.1 Detailed Description

The Utils class.

Author

Antoine de Roquemaurel

8.96.2 Member Function Documentation

8.96.2.1 QString Utils::String::firstLetterToUpper (QString *s*) [static]

firstLetterToUpper Put the first letter of a string in capslock

Parameters

s	The string to display

Returns

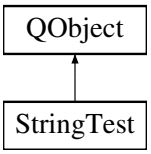
The new string with caps

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

- src/utils/string.h
- src/utils/string.cpp

8.97 StringTest Class Reference

Inheritance diagram for StringTest:



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

- tests/utils/stringtest.h
- tests/utils/stringtest.cpp

8.98 Mustache::Tag Struct Reference

`#include <mustache.h>`

Public Types

- enum Type {
  Null, **Value**, SectionStart, InvertedSectionStart,
  SectionEnd, Partial, Comment, SetDelimiter }
- enum **EscapeMode** { **Escape**, **Unescape**, **Raw** }

Public Attributes

- Type **type**
- QString **key**
- int **start**
- int **end**
- EscapeMode **escapeMode**

8.98.1 Detailed Description

Holds properties of a tag in a mustache template.

8.98.2 Member Enumeration Documentation

8.98.2.1 enum Mustache::Tag::Type

Enumerator

SectionStart A {{key}} or {{{key}}} tag.

InvertedSectionStart A {{#section}} tag.

SectionEnd An {{^inverted-section}} tag.

Partial A {{/section}} tag.

Comment A {{^partial}} tag.

SetDelimiter A {{! comment }} tag. A {{=<% %>=}} tag

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

- src/libs/qt-mustache/src/mustache.h

8.99 TestAdder< T > Class Template Reference

Public Member Functions

- **TestAdder** (const QString &name)

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

- tests/QTestRunner/testadder.cpp

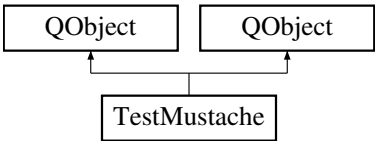
8.100 testadder Class Reference

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

- tests/QTestRunner/testadder.h

8.101 TestMustache Class Reference

Inheritance diagram for TestMustache:



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

- src/libs/qt-mustache/tests/test_mustache.h
- src/libs/qt-mustache/tests/test_mustache.cpp

8.102 TestRunner Class Reference

Public Member Functions

- template<typename T >
  char **RegisterTest** (QString name)
- int **RunAll** ()

Static Public Member Functions

- static TestRunner & **Instance** ()

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

- tests/QTestRunner/testrunner.h
- tests/QTestRunner/testrunner.cpp

8.103 Generator::TexGenerator Class Reference

The TexGenerator class Generate a LaTeX file.

`#include <texgenerator.h>`

Public Member Functions

- TexGenerator (QString tpl)

  TexGenerator::TexGenerator Construct a TexGenerator.
- void generate (QVariantHash data, QString path)

  TexGenerator::generate Generate a LaTeX file into a file specified by the path and which contains data

8.103.1 Detailed Description

The TexGenerator class Generate a LaTeX file.

8.103.2 Constructor & Destructor Documentation

8.103.2.1 Generator::TexGenerator::TexGenerator (QString *tpl*)

TexGenerator::TexGenerator Construct a TexGenerator.

Parameters

tpl	Template LaTeX to apply

8.103.3 Member Function Documentation

8.103.3.1 void Generator::TexGenerator::generate (QVariantHash *data*, QString *path*)

TexGenerator::generate Generate a LaTeX file into a file specified by the *path* and which contains *data*

Parameters

data	Data to integrate in the file
path	Path of the out file

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

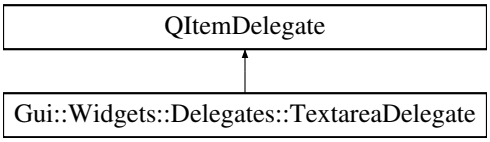
- src/generator/texgenerator.h
- src/generator/texgenerator.cpp

8.104 Gui::Widgets::Delegates::TextareaDelegate Class Reference

The TextareaDelegate class.

`#include <textareadelegate.h>`

Inheritance diagram for Gui::Widgets::Delegates::TextareaDelegate:



Public Member Functions

- TextareaDelegate (QWidget ∗parent=0)

  TextareaDelegate::TextareaDelegate.
- void setModelData (QWidget ∗editor, QAbstractItemModel ∗model, const QModelIndex &index) const

  TextareaDelegate::setModelData Sets the data model from content in editor.
- void updateEditorGeometry (QWidget ∗editor, const QStyleOptionViewItem &option, const QModelIndex &) const

  TextareaDelegate::updateEditorGeometry Update the editor for the item specified by index according to the style option given.
- QWidget ∗ createEditor (QWidget ∗parent, const QStyleOptionViewItem &, const QModelIndex &) const

  TextareaDelegate::createEditor Return a QTextEdit specified by index item defined by the parent widget and style option which are used to control how the editor widgets appears.
- void setEditorData (QWidget ∗editor, const QModelIndex &index) const

  TextareaDelegate::setEditorData Sets the data to be displayed and edited by the editor from the data model item specified by the model index

8.104.1 Detailed Description

The TextareaDelegate class.

Author

Antoine de Roquemaurel

8.104.2 Constructor & Destructor Documentation

8.104.2.1 Gui::Widgets::Delegates::TextareaDelegate::TextareaDelegate (QWidget * *parent* = 0)

TextareaDelegate::TextareaDelegate.

Parameters

parent	

8.104.3 Member Function Documentation

8.104.3.1 QWidget * Gui::Widgets::Delegates::TextareaDelegate::createEditor (QWidget * *parent*, const QStyleOptionViewItem & , const QModelIndex &) const

TextareaDelegate::createEditor Return a QTextEdit specified by *index* item defined by the *parent* widget and style *option* which are used to control how the editor widgets appears.

Parameters

parent	Widget parent
option	Option style
index	Index for editing

Returns

TextareaDelegate

8.104.3.2 void Gui::Widgets::Delegates::TextareaDelegate::setEditorData (QWidget * *editor*, const QModelIndex & *index*) const

TextareaDelegate::setEditorData Sets the data to be displayed and edited by the *editor* from the data model item specified by the model *index*

Parameters

editor	Data edited
index	Index of the model to edit

8.104.3.3 void Gui::Widgets::Delegates::TextareaDelegate::setModelData (QWidget * *editor*, QAbstractItemModel * *model*, const QModelIndex & *index*) const

TextareaDelegate::setModelData Sets the data model from content in editor.

Parameters

editor	Data edited
index	Index of the model to edit

8.104.3.4 void Gui::Widgets::Delegates::TextareaDelegate::updateEditorGeometry (QWidget * *editor,* const QStyleOptionViewItem & *option,* const QModelIndex &) const

TextareaDelegate::updateEditorGeometry Update the *editor* for the item specified by *index* according to the style *option* given.

Parameters

editor	Editor widget to update
option	Style option
index	Item index

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

- src/gui/widgets/delegates/textareadelegate.h
- src/gui/widgets/delegates/textareadelegate.cpp

8.105 Models::Unit Class Reference

The Unit class An unity for billing calculs.

`#include <unit.h>`

Public Member Functions

- Unit (TypeUnit type)

  Unit Construct an unit with init type.
- Unit ()

  Unit Construct an unit.
- QString toString (const bool plurial=false)

  toString Return String for unit type.
- TypeUnit getype () const

  getype The unit Type
- void setType (const TypeUnit &type)

  setType Change the unit Type
- bool operator== (const Unit &u)

  *operator == define the operator "==" to compare two **Unit***
- bool operator!= (const Unit &u)

  *operator != define the operator "!=" to compare two **Unit***

8.105.1 Detailed Description

The Unit class An unity for billing calculs.

8.105.2 Constructor & Destructor Documentation

8.105.2.1 Models::Unit::Unit (TypeUnit *type*)

Unit Construct an unit with init type.

Parameters

type	The unit Type;

8.105.3 Member Function Documentation

8.105.3.1 TypeUnit Models::Unit::getype ( ) const

getype The unit Type

Returns

The unit Type

8.105.3.2 bool Models::Unit::operator!= (const Unit & *u*)

operator != define the operator "!=" to compare two **Unit**

Parameters

c	the **Unit** to compare with the current **Contributory**

Returns

true if the **Unit** are different else false

8.105.3.3 bool Models::Unit::operator== (const Unit & *u*)

operator == define the operator "==" to compare two **Unit**

Parameters

c	the **Unit** to compare with the current **Contributory**

Returns

true if the **Unit** are equals else false

8.105.3.4 void Models::Unit::setType (const TypeUnit & *type*)

setType Change the unit Type

Parameters

type	The new unit type.

8.105.3.5 QString Models::Unit::toString (const bool *plurial* = `false`)

toString Return String for unit type.

Returns

The unit in String

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

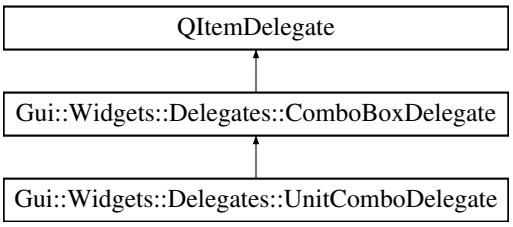
- src/models/unit.h
- src/models/unit.cpp

8.106 Gui::Widgets::Delegates::UnitComboDelegate Class Reference

The UnitComboDelegate class.

`#include <unitcombodelegate.h>`

Inheritance diagram for Gui::Widgets::Delegates::UnitComboDelegate:



Public Member Functions

- UnitComboDelegate (QObject ∗parent=0)

  UnitComboDelegate::UnitComboDelegate.
- QWidget ∗ createEditor (QWidget ∗parent, const QStyleOptionViewItem &option, const QModelIndex &index) const

  UnitComboDelegate::createEditor Return a ComboBox specified by index item defined by the parent widget and style option which are used to control how the editor widgets appears.
- void paint (QPainter ∗painter, const QStyleOptionViewItem &option, const QModelIndex &index) const

  UnitComboDelegate::paint Renders the delegate using the given painter and style option for the item specified by index

8.106.1 Detailed Description

The UnitComboDelegate class.

8.106.2 Constructor & Destructor Documentation

8.106.2.1 Gui::Widgets::Delegates::UnitComboDelegate::UnitComboDelegate (QObject ∗ *parent* = 0)

UnitComboDelegate::UnitComboDelegate.

Parameters

parent	

8.106.3 Member Function Documentation

8.106.3.1 QWidget ∗ Gui::Widgets::Delegates::UnitComboDelegate::createEditor (QWidget ∗ *parent,* const QStyleOptionViewItem & *option,* const QModelIndex & *index*) const [virtual]

UnitComboDelegate::createEditor Return a ComboBox specified by *index* item defined by the *parent* widget and style *option* which are used to control how the editor widgets appears.

Parameters

parent	Widget parent
option	Option style
index	Index for editing

Returns

ComboBox

Implements Gui::Widgets::Delegates::ComboBoxDelegate.

8.106.3.2 void Gui::Widgets::Delegates::UnitComboDelegate::paint (QPainter * *painter,* const QStyleOptionViewItem & *option,* const QModelIndex & *index*) const [virtual]

UnitComboDelegate::paint Renders the delegate using the given *painter* and style *option* for the item specified by *index*

Parameters

parent	Widget parent
option	Option style
index	Index for editing

Implements Gui::Widgets::Delegates::ComboBoxDelegate.

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

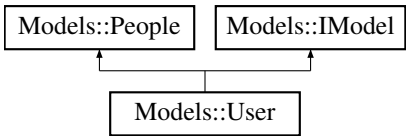
- src/gui/widgets/delegates/unitcombodelegate.h
- src/gui/widgets/delegates/unitcombodelegate.cpp

8.107 Models::User Class Reference

The User class **User** of it application.

`#include <user.h>`

Inheritance diagram for Models::User:



Public Member Functions

- User ()

  User::User. Contruct an User.
- User (int id)

  User::User. Construct a User with the identify id
- void commit ()

  User::commit Update user data in User table on the database.
- void hydrat (int id=1)

  User::hydrat Get data of the user who is specified by id from the database.
- void remove ()

  remove Remove the current User
- QVariantHash getDataMap ()

  getDataMap Get all data of model with a HashMap key/value
- void updateFolders (void)

  MainWindow::updateFolders Make directories which contain quotes and billings. Directories are the same than theirs of the Tree organisation (without Projects).
- QString getTitle () const

  User::getTitle Return a short description of User (company) activity.
- void setTitle (const QString &title)

  User::setTitle Modify the user/company activities description
- QString getNoSiret () const

User::getNoSiret Return the SIRET number (company registration number)

- void setNoSiret (const QString &noSiret)

  User::setNoSiret Modify the SIRET number (company registration number) noSiret

- QString getWorkspaceName () const

  User::getWorkspaceName Return the name of the workspace user.

- void setWorkspaceName (const QString &workspaceName)

  User::setWorkspaceName Change the current workspace name by the new workspaceName

- QString getWorkspacePath () const

  User::getWorkspacePath Return the path of the workspace user.

- void setWorkspacePath (const QString &workspacePath)

  User::setWorkspacePath Change the current workspace path by the new workspacePath

- bool operator== (const User &u)

  *User::operator == Re-define the operator "==" to compare if the current User is the same to the other **User** c Return TRUE if both Users are the same, else FALSE.*

- bool operator!= (const User &u)

  *User::operator == Re-define the operator "!=" to compare if the current User is differnt to the other **User** c Return TRUE if both Users are different, else FALSE.*

- QString getPdflatexPath () const

  getPdflatexPath The path of pdflatex

- void setPdflatexPath (const QString &getPdflatexPath)

  setPdflatexPath Change the pdflatex command path.

- int getNbHoursPerDays ()

  nbHoursPerDays The number of hours per days who the user work.

- int getNbDaysPerMonths ()

  getNbDaysPerMonths The number of days per months who the user work.

Additional Inherited Members

8.107.1 Detailed Description

The User class **User** of it application.

Author

Florent Berbie

8.107.2 Constructor & Destructor Documentation

8.107.2.1 Models::User::User (int *id*)

User::User. Construct a User with the identify *id*

Parameters

id	User id

8.107.3 Member Function Documentation

8.107.3.1 QVariantHash Models::User::getDataMap ( ) [virtual]

getDataMap Get all data of model with a HashMap key/value

Returns

Model's data

Implements Models::IModel.

8.107.3.2 int Models::User::getNbDaysPerMonths ( )

getNbDaysPerMonths The number of days per months who the user work.

Returns

The number of days per months

8.107.3.3 int Models::User::getNbHoursPerDays ( )

nbHoursPerDays The number of hours per days who the user work.

Returns

The number of hours per days

8.107.3.4 QString Models::User::getNoSiret ( ) const

User::getNoSiret Return the SIRET number (company registration number)

Returns

SIRET number

8.107.3.5 QString Models::User::getPdflatexPath ( ) const

getPdflatexPath The path of pdflatex

Returns

The pdflatex path

8.107.3.6 QString Models::User::getTitle ( ) const

User::getTitle Return a short description of User (company) activity.

Returns

a short description of user (company) activity

8.107.3.7 QString Models::User::getWorkspaceName ( ) const

User::getWorkspaceName Return the name of the workspace user.

Returns

workspace name

8.107.3.8 QString Models::User::getWorkspacePath ( ) const

User::getWorkspacePath Return the path of the workspace user.

Returns

workspace path

8.107.3.9 void Models::User::hydrat (int *id* = 1) [virtual]

User::hydrat Get data of the user who is specified by *id* from the database.

Parameters

id	User identify

Implements Models::IModel.

8.107.3.10 bool Models::User::operator!= (const User & *u*)

User::operator == Re-define the operator "!=" to compare if the current User is differnt to the other **User** *c* Return TRUE if both Users are different, else FALSE.

Parameters

u	User to compare

Returns

boolean

8.107.3.11 bool Models::User::operator== (const User & *u*)

User::operator == Re-define the operator "==" to compare if the current User is the same to the other **User** *c* Return TRUE if both Users are the same, else FALSE.

Parameters

u	User to compare

Returns

boolean

8.107.3.12 void Models::User::setNoSiret (const QString & *noSiret*)

User::setNoSiret Modify the SIRET number (company registration number) *noSiret*

Parameters

noSiret	SIRET number

8.107.3.13 void Models::User::setPdflatexPath (const QString & *getPdflatexPath*)

setPdflatexPath Change the pdflatex command path.

Parameters

getPdflatexPath	The new command

8.107.3.14 void Models::User::setTitle (const QString & *title*)

User::setTitle Modify the user/company activities *description*

Parameters

title	Short description on activity(ies) of User company

8.107.3.15 void Models::User::setWorkspaceName (const QString & *workspaceName*)

User::setWorkspaceName Change the current workspace name by the new *workspaceName*

Parameters

workspaceName	

8.107.3.16 void Models::User::setWorkspacePath (const QString & *workspacePath*)

User::setWorkspacePath Change the current workspace path by the new *workspacePath*

Parameters

workspacePath	

8.107.3.17 void Models::User::updateFolders (void)

MainWindow::updateFolders Make directories which contain quotes and billings. Directories are the same than theirs of the Tree organisation (without Projects).

Organisation of folders are formed like this:

- COMPANY CustomerLastname CustomerFirstname/
  - Quotes/
    - quote1 ...
  - Billings/
    - billing1 ...

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

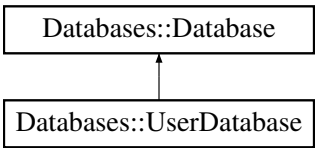
- src/models/user.h
- src/models/user.cpp

8.108 Databases::UserDatabase Class Reference

The UserDatabase class Access to User data in the the table User of the **Database**

`#include <userdatabase.h>`

Inheritance diagram for Databases::UserDatabase:



Public Member Functions

- Models::User * getUser (const int pId=1)

  UserDatabase::getUser Get informations about the user (identified by 'pId')
- void updateUser (const Models::User &)

  UserDatabase::updateUser Update informations about the user.

Static Public Member Functions

- static UserDatabase * instance () throw (DbException*)

  UserDatabase::getInstance Return an instance of UserDatabase.

Additional Inherited Members

8.108.1 Detailed Description

The UserDatabase class Access to User data in the the table User of the **Database**

Author

Florent Berbie

See also

Database
User

8.108.2 Member Function Documentation

8.108.2.1 Models::User * Databases::UserDatabase::getUser (const int *pId* = `1`)

UserDatabase::getUser Get informations about the user (identified by 'pId')

Parameters

pId	user id (1 default)

Returns

the user

8.108.2.2 UserDatabase * Databases::UserDatabase::instance ( ) throw DbException *) `[static]`

UserDatabase::getInstance Return an instance of UserDatabase.

Returns

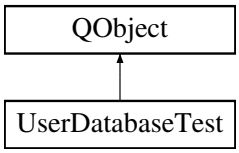
Instance of UserDatabase

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

- src/database/userdatabase.h
- src/database/userdatabase.cpp

8.109 UserDatabaseTest Class Reference

Inheritance diagram for UserDatabaseTest:



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

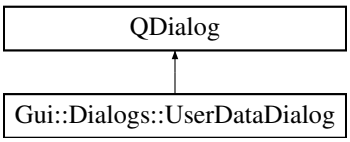
- tests/database/userdatabasetest.h
- tests/database/userdatabasetest.cpp

8.110 Gui::Dialogs::UserDataDialog Class Reference

The UserDataDialog class Window to fill user data.

`#include <userdatadialog.h>`

Inheritance diagram for Gui::Dialogs::UserDataDialog:



Public Slots

- void checkFields ()

  UserDataDialog::checkFields Check all fields of dialog components.

Public Member Functions

- UserDataDialog (QWidget *parent=0)

  UserDataDialog::UserDataDialog Construct a window with user data.
- void fillFields ()

  UserDataDialog::fillFields Fill line edits with the data of the user.
- void accept ()

  UserDataDialog::accept Valid data inputed by user and add these data in Database.
- void reject ()

  UserDataDialog::reject Cancel the operation and close the windows.

8.110.1 Detailed Description

The UserDataDialog class Window to fill user data.

Author

Florent Berbie

See also

Project

8.110.2 Constructor & Destructor Documentation

8.110.2.1 Gui::Dialogs::UserDataDialog::UserDataDialog (QWidget * *parent* = 0) [explicit]

UserDataDialog::UserDataDialog Construct a window with user data.

Parameters

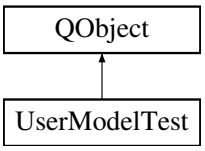
parent	

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

- src/gui/dialogs/userdatadialog.h
- src/gui/dialogs/userdatadialog.cpp

8.111 UserModelTest Class Reference

Inheritance diagram for UserModelTest:



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

- tests/models/usermodeltest.h
- tests/models/usermodeltest.cpp

Index

add
    Gui::Widgets::ContributoriesWidget, 83
addBill
    Gui::MainWindow, 129
addBilling
    Databases::BillingDatabase, 33
addBillingProject
    Databases::BillingDatabase, 33
addBillingToProject
    Utils::HierarchicalSystem, 120
addContributory
    Databases::ContributoryDatabase, 89
    Models::Billing, 27
    Models::ContributoriesList, 75
addCustomer
    Databases::CustomerDatabase, 97
addDoc
    Gui::MainWindow, 129
addProject
    Databases::ProjectDatabase, 156
    Gui::MainWindow, 130
    Gui::Widgets::ContributoriesWidget, 83
    Models::ContributoriesList, 76
AddProjectDialog
    Gui::Dialogs::AddProjectDialog, 22
addProjectToCustomer
    Utils::HierarchicalSystem, 120
addQuote
    Gui::MainWindow, 130
AddQuoteDialog
    Gui::Dialogs::AddQuoteDialog, 23
addRateProject
    Databases::RateDatabase, 169
allProjectsChosen
    Gui::Widgets::WdgModels::ProjectContributories↩
        TableModel, 152
append
    Gui::Widgets::WdgModels::BillingsTableModel, 40
    Gui::Widgets::WdgModels::ContributoriesTable↩
        Model, 80
    Gui::Widgets::WdgModels::CustomersTableModel,
        103
    Gui::Widgets::WdgModels::ProjectContributories↩
        TableModel, 152
    Gui::Widgets::WdgModels::ProjectsTableModel,
        161

beginDateControl
    Gui::Dialogs::ComputeTurnoverDialog, 70
Billing
    Models::Billing, 27
BillingDatabaseTest, 37
BillingModelTest, 37
BillingsTableModel
    Gui::Widgets::WdgModels::BillingsTableModel, 39

canEval
    CounterContext, 92
    Mustache::Context, 72
    Mustache::QtVariantContext, 165
changeDatabase
    Databases::Database, 108
check
    Gui::Widgets::CheckFields::CheckEmail, 45
    Gui::Widgets::CheckFields::CheckFieldsLetters, 46
    Gui::Widgets::CheckFields::CheckFieldsNumbers,
        48
    Gui::Widgets::CheckFields::CheckIpAddress, 49
    Gui::Widgets::CheckFields::CheckLogin, 50
    Gui::Widgets::CheckFields::CheckPhone, 52
    Gui::Widgets::CheckFields::CheckPortNumber, 54
    Gui::Widgets::CheckFields::CheckPostalCode, 55
    Gui::Widgets::CheckFields::CheckSiretNumber, 58
    Gui::Widgets::CheckFields::CheckUntilField, 59
    Gui::Widgets::CheckFields::CheckValidField, 60
    Gui::Widgets::CheckFields::CheckWebsite, 61
    Gui::Widgets::CheckFields::ICheckField, 121
CheckCity
    Gui::Widgets::CheckFields::CheckCity, 43
CheckCountry
    Gui::Widgets::CheckFields::CheckCountry, 44
CheckEmail
    Gui::Widgets::CheckFields::CheckEmail, 45
CheckFieldsLetters
    Gui::Widgets::CheckFields::CheckFieldsLetters, 46
CheckFieldsNumbers
    Gui::Widgets::CheckFields::CheckFieldsNumbers,
        47
CheckIpAddress
    Gui::Widgets::CheckFields::CheckIpAddress, 49
CheckLogin
    Gui::Widgets::CheckFields::CheckLogin, 50
CheckName
    Gui::Widgets::CheckFields::CheckName, 51
CheckPhone
    Gui::Widgets::CheckFields::CheckPhone, 52
CheckPortNumber
    Gui::Widgets::CheckFields::CheckPortNumber, 54
CheckPostalCode
    Gui::Widgets::CheckFields::CheckPostalCode, 55

CheckQLineEdit
Gui::Widgets::CheckFields::CheckQLineEdit, 56
CheckSiretNumber
Gui::Widgets::CheckFields::CheckSiretNumber, 58
CheckUntilField
Gui::Widgets::CheckFields::CheckUntilField, 59
CheckValidField
Gui::Widgets::CheckFields::CheckValidField, 60
CheckWebsite
Gui::Widgets::CheckFields::CheckWebsite, 61
ChoseDirectoryWidget
Gui::Widgets::Path::ChoseDirectoryWidget, 62
ChoseFileWidget
Gui::Widgets::Path::ChoseFileWidget, 63
ChosePathWidget
Gui::Widgets::Path::ChosePathWidget, 65
columnCount
Gui::Widgets::WdgModels::BillingsTableModel, 40
Gui::Widgets::WdgModels::ContributoriesTable↩
Model, 80
Gui::Widgets::WdgModels::CustomersTableModel,
104
Gui::Widgets::WdgModels::ProjectContributories↩
TableModel, 152
Gui::Widgets::WdgModels::ProjectsTableModel,
161
ComboBoxDelegate
Gui::Widgets::Delegates::ComboBoxDelegate, 67
ComboBoxModelWidget
Gui::Widgets::ComboBoxModelWidget, 69
Comment
Mustache::Tag, 187
Context
Mustache::Context, 72
ContributoriesDatabaseTest, 73
ContributoriesTableModel
Gui::Widgets::WdgModels::ContributoriesTable↩
Model, 79
ContributoriesWidget
Gui::Widgets::ContributoriesWidget, 83
Contributory
Models::Contributory, 85
ContributoryListTest, 91
ContributoryModelTest, 91
count
Gui::Widgets::ContributoriesWidget, 83
Gui::Widgets::WdgModels::BillingsTableModel, 40
Gui::Widgets::WdgModels::ContributoriesTable↩
Model, 80
Gui::Widgets::WdgModels::CustomersTableModel,
104
Gui::Widgets::WdgModels::ProjectsTableModel,
161
CounterContext, 92
canEval, 92
eval, 92, 93
stringValue, 93
createEditor
Gui::Widgets::Delegates::ComboBoxDelegate, 67
Gui::Widgets::Delegates::DoubleSpinBoxDelegate,
116
Gui::Widgets::Delegates::ProjectComboDelegate,
150
Gui::Widgets::Delegates::TextareaDelegate, 191
Gui::Widgets::Delegates::UnitComboDelegate, 195
Customer
Models::Customer, 94
CustomerContextualMenu
Gui::Widgets::CustomerContextualMenu, 96
CustomerDataWidget
Gui::Widgets::CustomerDataWidget, 101
CustomerDatabaseTest, 101
CustomerModelTest, 102
CustomersTableModel
Gui::Widgets::WdgModels::CustomersTableModel,
103

DB_FILENAME
Parameters, 133
data
Gui::Widgets::WdgModels::BillingsTableModel, 40
Gui::Widgets::WdgModels::ContributoriesTable↩
Model, 80
Gui::Widgets::WdgModels::CustomersTableModel,
104
Gui::Widgets::WdgModels::ProjectContributories↩
TableModel, 153
Gui::Widgets::WdgModels::ProjectsTableModel,
161
DatabaseSettingsWidget
Gui::Widgets::DatabaseSettingsWidget, 110
databaseTypeChanged
Gui::Dialogs::StartedWindowsDialog, 184
Databases, 17
Databases::AccessDatabase, 21
Databases::BillingDatabase, 31
addBilling, 33
addBillingProject, 33
getAllBillingsOfProject, 33
getAllBillingsOnly, 33
getBilling, 33
getBillings, 34
getBillingsBetweenDates, 34
getBillingsTable, 34
getMaxBillingNumber, 34
getMaxBillingNumberOfCustomer, 34
getMaxQuoteNumber, 36
getMaxQuoteNumberOfCustomer, 36
instance, 36
isBillingPaid, 36
removeBilling, 36
removeBillingProject, 37
Databases::ContributoryDatabase, 88
addContributory, 89
getContributoriesByBilling, 89
getContributoriesByBillingAndProject, 90
getContributory, 90

instance, 90
removeContributory, 91
Databases::CustomerDatabase, 96
addCustomer, 97
getCustomer, 97
getCustomersTable, 99
getItemCustomer, 99
getItemProject, 99
getItemRoot, 99
getNbCustomers, 99
getTree, 100
instance, 100
removeCustomer, 100
updateCustomer, 100
Databases::Database, 106
changeDatabase, 108
executeFile, 108
instance, 108
lastError, 108
setDatabase, 109
value, 109
Databases::ProjectDatabase, 155
addProject, 156
getAllProjects, 156
getCostProjects, 156
getNbProjects, 156
getNbProjectsForACustomer, 156
getProject, 157
getProjects, 157
getProjectsOfCustomer, 157
getProjectsTable, 157
instance, 159
removeProject, 159
updateProject, 159
Databases::RateDatabase, 168
addRateProject, 169
getRate, 169
instance, 169
updateRateProject, 169
Databases::UserDatabase, 200
getUser, 201
instance, 201
DbException
Exceptions::DbException, 112
DialogAddCustomer
Gui::Dialogs::DialogAddCustomer, 113
DoubleSpinBoxDelegate
Gui::Widgets::Delegates::DoubleSpinBoxDelegate, 116

editUser
Gui::MainWindow, 130
endDateControl
Gui::Dialogs::ComputeTurnoverDialog, 71
error
Mustache::Renderer, 172
errorPartial
Mustache::Renderer, 172
errorPos
Mustache::Renderer, 172
eval
CounterContext, 92, 93
Mustache::Context, 72
Mustache::QtVariantContext, 165
Exceptions::DbException, 111
DbException, 112
popupMessage, 112
Exceptions::FileException, 117
FileException, 118
popupMessage, 119
executeFile
Databases::Database, 108

FileChoseWidget, 117
FileException
Exceptions::FileException, 118
fillLabels
Gui::Dialogs::ComputeTurnoverDialog, 71
fillQuoteBilling
Gui::Dialogs::AddQuoteDialog, 24
filterOnBillsOrQuotes
Models::Search, 174
filterOnCompany
Models::Search, 174
filterOnContributories
Models::Search, 174
filterOnCustomersWithoutProject
Models::Search, 175
filterOnNumberElements
Models::Search, 175
filterOnProjects
Models::Search, 175
filterOnReferentLastname
Models::Search, 175
filterOnVarcharElements
Models::Search, 175
firstLetterToUpper
Utils::String, 186
flags
Gui::Widgets::WdgModels::BillingsTableModel, 40
Gui::Widgets::WdgModels::ContributoriesTableModel, 80
Gui::Widgets::WdgModels::CustomersTableModel, 104
Gui::Widgets::WdgModels::ProjectContributoriesTableModel, 153
Gui::Widgets::WdgModels::ProjectsTableModel, 162
fn_t
Mustache::QtVariantContext, 165

generate
Generator::PdfGenerator, 136
Generator::TexGenerator, 190
Generation, 119
Generator::PdfGenerator, 136
generate, 136
PdfGenerator, 136

Generator::TexGenerator, 188
generate, 190
TexGenerator, 189
getAddress
Models::People, 138
getAddressComplement
Models::People, 138
getAllBillingsOfProject
Databases::BillingDatabase, 33
getAllBillingsOnly
Databases::BillingDatabase, 33
getAllContributories
Models::ContributoriesList, 76
getAllProjects
Databases::ProjectDatabase, 156
getBeginDate
Models::Project, 145
getBilling
Databases::BillingDatabase, 33
getBillings
Databases::BillingDatabase, 34
Gui::Widgets::WdgModels::BillingsTableModel, 40
getBillingsBetweenDates
Databases::BillingDatabase, 34
getBillingsTable
Databases::BillingDatabase, 34
getBtnValid
Gui::Widgets::CheckFields::CheckQLineEdit, 57
getCity
Models::People, 138
getCompany
Models::People, 138
getContributories
Gui::Widgets::ContributoriesWidget, 83
Gui::Widgets::WdgModels::ContributoriesTable↩
Model, 80
Models::Billing, 27
Models::ContributoriesList, 76
getContributoriesByBilling
Databases::ContributoryDatabase, 89
getContributoriesByBillingAndProject
Databases::ContributoryDatabase, 90
getContributory
Databases::ContributoryDatabase, 90
getCopy
Gui::Dialogs::AddQuoteDialog, 24
getCost
Models::Project, 145, 146
getCostProjects
Databases::ProjectDatabase, 156
getCountry
Gui::Widgets::CheckFields::CheckPhone, 52
Gui::Widgets::CheckFields::CheckPostalCode, 55
Models::People, 139
getCurrentCustomerId
Gui::MainWindow, 130
Gui::Widgets::searchWidget, 182
getCurrentCustomerName
Gui::MainWindow, 130
getCurrentProjectId
Gui::MainWindow, 130
getCurrentProjectName
Gui::MainWindow, 130
getCurrentQuoteId
Gui::MainWindow, 131
getCustomer
Databases::CustomerDatabase, 97
Models::ContributoriesList, 76
Models::Project, 146
getCustomers
Gui::Widgets::WdgModels::CustomersTableModel,
104
Utils::HierarchicalSystem, 120
getCustomersTable
Databases::CustomerDatabase, 99
getDailyRate
Gui::Widgets::RateWidget, 171
Models::Project, 146
Models::Rate, 167
getDataMap
Models::Billing, 27
Models::ContributoriesList, 76
Models::Contributory, 85
Models::Customer, 94
Models::IModel, 123
Models::Project, 146
Models::User, 197
getDatabaseName
Gui::Widgets::DatabaseSettingsWidget, 110
getDate
Models::Billing, 27
getDefaultLocation
Gui::Widgets::Path::ChosePathWidget, 65
getDescription
Models::Billing, 27
Models::Contributory, 85
Models::Project, 146
getDomainNameOrIP
Gui::Widgets::DatabaseSettingsWidget, 110
getEmail
Models::People, 139
getEndDate
Models::Project, 146
getFax
Models::People, 139
getField
Gui::Widgets::Path::ChosePathWidget, 65
getFilename
Models::Billing, 28
getFilter
Models::Search, 175
getFirstname
Models::People, 139
getFolder
Models::Billing, 28
getGroupFilter

Models::Search, 176
getHourlyRate
Gui::Widgets::RateWidget, 171
Models::Contributory, 86
Models::Rate, 167
getId
Models::IModel, 123
getIdBilling
Models::ContributoriesList, 76
getIdCustomer
Gui::Dialogs::AddQuoteDialog, 24
Gui::Widgets::searchWidget, 182
getImage
Gui::Dialogs::StartedWindowsDialog, 184
getItem
Models::Billing, 28
getItemCustomer
Databases::CustomerDatabase, 99
getItemProject
Databases::CustomerDatabase, 99
getItemRoot
Databases::CustomerDatabase, 99
getLastname
Models::People, 139
getLegalRate
Models::Rate, 168
getLogin
Gui::Widgets::DatabaseSettingsWidget, 111
getLongDescription
Models::Contributory, 86
getMaxBillingNumber
Databases::BillingDatabase, 34
getMaxBillingNumberOfCustomer
Databases::BillingDatabase, 34
getMaxQuoteNumber
Databases::BillingDatabase, 36
getMaxQuoteNumberOfCustomer
Databases::BillingDatabase, 36
getMobilePhone
Models::People, 139
getModel
Utils::ItemType, 125
getName
Models::Project, 147
Utils::ItemType, 125
getNameFolder
Models::Customer, 94
getNbCustomers
Databases::CustomerDatabase, 99
getNbDailyHours
Models::Rate, 168
getNbDaysPerMonths
Models::User, 198
getNbHoursPerDays
Models::User, 198
getNbProjects
Databases::ProjectDatabase, 156
Models::ContributoriesList, 76
getNbProjectsForACustomer
Databases::ProjectDatabase, 156
getNoSiret
Models::User, 198
getNumber
Gui::Dialogs::AddQuoteDialog, 24
Models::Billing, 28
getPartial
Mustache::PartialFileLoader, 134
Mustache::PartialMap, 135
Mustache::PartialResolver, 136
getPassword
Gui::Widgets::DatabaseSettingsWidget, 111
getPath
Models::Billing, 28
Models::Customer, 94
getPdflatexPath
Models::User, 198
getPhone
Models::People, 139
getPort
Gui::Widgets::DatabaseSettingsWidget, 111
getPostalCode
Models::People, 140
getPrice
Models::Billing, 28
Models::Calculable, 42
Models::ContributoriesList, 77
Models::Contributory, 86
getProject
Databases::ProjectDatabase, 157
Gui::Widgets::WdgModels::ProjectContributories↩
TableModel, 153
Models::Contributory, 86
getProjects
Databases::ProjectDatabase, 157
Gui::Widgets::Delegates::ProjectComboDelegate,
150
Gui::Widgets::WdgModels::ProjectContributories↩
TableModel, 153
Gui::Widgets::WdgModels::ProjectsTableModel,
162
Models::ContributoriesList, 77
Utils::HierarchicalSystem, 120
getProjectsOfCustomer
Databases::ProjectDatabase, 157
getProjectsTable
Databases::ProjectDatabase, 157
getQuantity
Models::Contributory, 86
getRate
Databases::RateDatabase, 169
Models::ContributoriesList, 77
getSearchInBillsQuotes
Models::Search, 176
getSearchInCompanies
Models::Search, 176
getSearchInProjects

Models::Search, 176
getSearchInReferentLastname
Models::Search, 176
getSelectedProjects
Gui::Widgets::WdgModels::ProjectContributories↩
TableModel, 153
getSumQuantity
Gui::Widgets::WdgModels::ContributoriesTable↩
Model, 81
Models::Billing, 29
Models::Calculable, 42
Models::ContributoriesList, 77
Models::Contributory, 86
getText
Models::Search, 176
getTitle
Models::Billing, 29
Models::User, 198
getTree
Databases::CustomerDatabase, 100
getTurnover
Models::Customer, 95
getTurnoverBetweenDates
Models::Statistics, 185
getType
Utils::ItemType, 125
getUnit
Models::Contributory, 87
getUser
Databases::UserDatabase, 201
getWebsite
Models::People, 140
getWorkspaceName
Models::User, 198
getWorkspacePath
Models::User, 198
getype
Models::Unit, 193
getypeFiles
Gui::Widgets::Path::ChoseFileWidget, 64
Gui, 17
Gui::Dialogs::AddProjectDialog, 21
AddProjectDialog, 22
Gui::Dialogs::AddQuoteDialog, 22
AddQuoteDialog, 23
fillQuoteBilling, 24
getCopy, 24
getIdCustomer, 24
getNumber, 24
setCopy, 24
setQuoteIdNumber, 25
Gui::Dialogs::ComputeTurnoverDialog, 70
beginDateControl, 70
endDateControl, 71
fillLabels, 71
Gui::Dialogs::DialogAddCustomer, 113
DialogAddCustomer, 113
Gui::Dialogs::MessageBox, 132
MessageBox, 132
setImage, 133
setText, 133
Gui::Dialogs::StartedWindowsDialog, 183
databaseTypeChanged, 184
getImage, 184
isDatabaseCentralized, 184
isDatabaseTypeValid, 185
StartedWindowsDialog, 184
Gui::Dialogs::UserDataDialog, 202
UserDataDialog, 203
Gui::Docks::SearchDock, 178
search, 180
SearchDock, 179
textChanged, 180
Gui::MainWindow, 127
addBill, 129
addDoc, 129
addProject, 130
addQuote, 130
editUser, 130
getCurrentCustomerId, 130
getCurrentCustomerName, 130
getCurrentProjectId, 130
getCurrentProjectName, 130
getCurrentQuoteId, 131
MainWindow, 129
resizeEvent, 131
rootTree, 131
search, 131
treeLevel, 131
Gui::Widgets, 18
Gui::Widgets::CheckFields::CheckCity, 43
CheckCity, 43
Gui::Widgets::CheckFields::CheckCountry, 43
CheckCountry, 44
Gui::Widgets::CheckFields::CheckEmail, 44
check, 45
CheckEmail, 45
Gui::Widgets::CheckFields::CheckFieldsLetters, 46
check, 46
CheckFieldsLetters, 46
Gui::Widgets::CheckFields::CheckFieldsNumbers, 47
check, 48
CheckFieldsNumbers, 47
Gui::Widgets::CheckFields::CheckIpAddress, 48
check, 49
CheckIpAddress, 49
Gui::Widgets::CheckFields::CheckLogin, 49
check, 50
CheckLogin, 50
Gui::Widgets::CheckFields::CheckName, 51
CheckName, 51
Gui::Widgets::CheckFields::CheckPhone, 51
check, 52
CheckPhone, 52
getCountry, 52
setCountry, 53

Gui::Widgets::CheckFields::CheckPortNumber, 53
check, 54
CheckPortNumber, 54
Gui::Widgets::CheckFields::CheckPostalCode, 54
check, 55
CheckPostalCode, 55
getCountry, 55
setCountry, 55
Gui::Widgets::CheckFields::CheckQLineEdit, 56
CheckQLineEdit, 56
getBtnValid, 57
isValid, 57
setBtnValid, 57
Gui::Widgets::CheckFields::CheckSiretNumber, 57
check, 58
CheckSiretNumber, 58
Gui::Widgets::CheckFields::CheckUntilField, 58
check, 59
CheckUntilField, 59
Gui::Widgets::CheckFields::CheckValidField, 59
check, 60
CheckValidField, 60
Gui::Widgets::CheckFields::CheckWebsite, 61
check, 61
CheckWebsite, 61
Gui::Widgets::CheckFields::ICheckField, 121
check, 121
Gui::Widgets::ComboBoxModelWidget, 69
ComboBoxModelWidget, 69
Gui::Widgets::ContributoriesWidget, 82
add, 83
addProject, 83
ContributoriesWidget, 83
count, 83
getContributories, 83
Gui::Widgets::CustomerContextualMenu, 95
CustomerContextualMenu, 96
Gui::Widgets::CustomerDataWidget, 101
CustomerDataWidget, 101
printInformations, 102
Gui::Widgets::DatabaseSettingsWidget, 109
DatabaseSettingsWidget, 110
getDatabaseName, 110
getDomainNameOrIP, 110
getLogin, 111
getPassword, 111
getPort, 111
isValid, 111
Gui::Widgets::Delegates::ComboBoxDelegate, 66
ComboBoxDelegate, 67
createEditor, 67
paint, 67
setEditorData, 67
setModelData, 67
updateEditorGeometry, 69
Gui::Widgets::Delegates::DoubleSpinBoxDelegate, 115
createEditor, 116
DoubleSpinBoxDelegate, 116
setEditorData, 116
setModelData, 116
updateEditorGeometry, 116
Gui::Widgets::Delegates::ProjectComboDelegate, 149
createEditor, 150
getProjects, 150
isLocked, 150
paint, 150
ProjectComboDelegate, 150
setLocked, 151
Gui::Widgets::Delegates::TextareaDelegate, 190
createEditor, 191
setEditorData, 191
setModelData, 191
TextareaDelegate, 191
updateEditorGeometry, 191
Gui::Widgets::Delegates::UnitComboDelegate, 194
createEditor, 195
paint, 195
UnitComboDelegate, 195
Gui::Widgets::Path::ChoseDirectoryWidget, 62
ChoseDirectoryWidget, 62
Gui::Widgets::Path::ChoseFileWidget, 63
ChoseFileWidget, 63
getypeFiles, 64
setTypeFiles, 64
Gui::Widgets::Path::ChosePathWidget, 64
ChosePathWidget, 65
getDefaultLocation, 65
getField, 65
setField, 65
Gui::Widgets::Popup, 143
Gui::Widgets::ProjectsWidget, 163
ProjectsWidget, 164
updateBtn, 164
Gui::Widgets::RateWidget, 170
getDailyRate, 171
getHourlyRate, 171
RateWidget, 171
setDailyRate, 171
setHourlyRate, 171
setWidgetDailyRateValue, 171
setWidgetHourlyRateValue, 172
Gui::Widgets::WdgModels::BillingsTableModel, 38
append, 40
BillingsTableModel, 39
columnCount, 40
count, 40
data, 40
flags, 40
getBillings, 40
headerData, 41
remove, 41
rowCount, 41
setData, 41
Gui::Widgets::WdgModels::ContributoriesTableModel, 78
append, 80

columnCount, 80
ContributoriesTableModel, 79
count, 80
data, 80
flags, 80
getContributories, 80
getSumQuantity, 81
headerData, 81
remove, 81
rowCount, 81
setData, 81
Gui::Widgets::WdgModels::CustomersTableModel, 102
append, 103
columnCount, 104
count, 104
CustomersTableModel, 103
data, 104
flags, 104
getCustomers, 104
headerData, 104
remove, 106
rowCount, 106
setData, 106
Gui::Widgets::WdgModels::ProjectContributories↩ TableModel, 151
allProjectsChosen, 152
append, 152
columnCount, 152
data, 153
flags, 153
getProject, 153
getProjects, 153
getSelectedProjects, 153
headerData, 154
ProjectContributoriesTableModel, 152
remove, 154
rowCount, 154
setData, 154
Gui::Widgets::WdgModels::ProjectsTableModel, 160
append, 161
columnCount, 161
count, 161
data, 161
flags, 162
getProjects, 162
headerData, 162
ProjectsTableModel, 161
remove, 162
rowCount, 162
setData, 163
Gui::Widgets::searchWidget, 180
getCurrentCustomerId, 182
getIdCustomer, 182
isCustomerSelected, 182
search, 182
searchWidget, 181
selectCustomer, 182
setIdCustomer, 182
headerData
Gui::Widgets::WdgModels::BillingsTableModel, 41
Gui::Widgets::WdgModels::ContributoriesTable↩ Model, 81
Gui::Widgets::WdgModels::CustomersTableModel, 104
Gui::Widgets::WdgModels::ProjectContributories↩ TableModel, 154
Gui::Widgets::WdgModels::ProjectsTableModel, 162
hydrat
Models::Billing, 29
Models::Contributory, 87
Models::Customer, 95
Models::IModel, 123
Models::Project, 147
Models::User, 199

instance
Databases::BillingDatabase, 36
Databases::ContributoryDatabase, 90
Databases::CustomerDatabase, 100
Databases::Database, 108
Databases::ProjectDatabase, 159
Databases::RateDatabase, 169
Databases::UserDatabase, 201
Utils::Log, 126
InvertedSectionStart
Mustache::Tag, 187
isBilling
Models::Billing, 29
isBillingPaid
Databases::BillingDatabase, 36
isCustomerSelected
Gui::Widgets::searchWidget, 182
isDatabaseCentralized
Gui::Dialogs::StartedWindowsDialog, 184
isDatabaseTypeValid
Gui::Dialogs::StartedWindowsDialog, 185
isFalse
Mustache::Context, 72
Mustache::QtVariantContext, 166
isInsert
Models::ContributoriesList, 78
isLocked
Gui::Widgets::Delegates::ProjectComboDelegate, 150
isPaid
Models::Billing, 29
isToRemoved
Models::IModel, 123
isValid
Gui::Widgets::CheckFields::CheckQLineEdit, 57
Gui::Widgets::DatabaseSettingsWidget, 111
ItemType
Utils::ItemType, 124
ItemTypeTest, 126

lastError

Databases::Database, 108
listCount
Mustache::Context, 72
Mustache::QtVariantContext, 166

MainWindow
Gui::MainWindow, 129
makeDirectory
Utils::Directories, 114
MessageBox
Gui::Dialogs::MessageBox, 132
Models, 18
Models::Billing, 25
addContributory, 27
Billing, 27
getContributories, 27
getDataMap, 27
getDate, 27
getDescription, 27
getFilename, 28
getFolder, 28
getItem, 28
getNumber, 28
getPath, 28
getPrice, 28
getSumQuantity, 29
getTitle, 29
hydrat, 29
isBilling, 29
isPaid, 29
operator!=, 29
operator<, 30
operator==, 30
setDate, 30
setDescription, 30
setIsBilling, 30
setIsPaid, 31
setNumber, 31
setTitle, 31
Models::Calculable, 42
getPrice, 42
getSumQuantity, 42
Models::ContributoriesList, 73
addContributory, 75
addProject, 76
getAllContributories, 76
getContributories, 76
getCustomer, 76
getDataMap, 76
getIdBilling, 76
getNbProjects, 76
getPrice, 77
getProjects, 77
getRate, 77
getSumQuantity, 77
isInsert, 78
setAllIdContributories, 78
setIdBilling, 78
setInsert, 78
Models::Contributory, 84
Contributory, 85
getDataMap, 85
getDescription, 85
getHourlyRate, 86
getLongDescription, 86
getPrice, 86
getProject, 86
getQuantity, 86
getSumQuantity, 86
getUnit, 87
hydrat, 87
operator!=, 87
operator==, 87
setDescription, 87
setHourlyRate, 88
setLongDescription, 88
setProject, 88
setQuantity, 88
setUnit, 88
Models::Customer, 93
Customer, 94
getDataMap, 94
getNameFolder, 94
getPath, 94
getTurnover, 95
hydrat, 95
Models::IModel, 122
getDataMap, 123
getId, 123
hydrat, 123
isToRemoved, 123
setId, 123
setToRemoved, 123
Models::People, 137
getAddress, 138
getAddressComplement, 138
getCity, 138
getCompany, 138
getCountry, 139
getEmail, 139
getFax, 139
getFirstname, 139
getLastname, 139
getMobilePhone, 139
getPhone, 139
getPostalCode, 140
getWebsite, 140
operator!=, 140
operator==, 140
setAddress, 140
setAddressComplement, 141
setCity, 141
setCompany, 141
setCountry, 141
setEmail, 141
setFax, 141
setFirstname, 142

setLastname, 142
setMobilePhone, 142
setPhone, 142
setPostalCode, 142
setWebsite, 142
Models::Project, 144
getBeginDate, 145
getCost, 145, 146
getCustomer, 146
getDailyRate, 146
getDataMap, 146
getDescription, 146
getEndDate, 146
getName, 147
hydrat, 147
operator!=, 147
operator<, 147
operator==, 147
Project, 145
setBeginDate, 148
setCost, 148
setCustomer, 148
setDailyRate, 148
setDescription, 148
setEndDate, 148
setName, 149
Models::Rate, 166
getDailyRate, 167
getHourlyRate, 167
getLegalRate, 168
getNbDailyHours, 168
Rate, 167
Models::Search, 173
filterOnBillsOrQuotes, 174
filterOnCompany, 174
filterOnContributories, 174
filterOnCustomersWithoutProject, 175
filterOnNumberElements, 175
filterOnProjects, 175
filterOnReferentLastname, 175
filterOnVarcharElements, 175
getFilter, 175
getGroupFilter, 176
getSearchInBillsQuotes, 176
getSearchInCompanies, 176
getSearchInProjects, 176
getSearchInReferentLastname, 176
getText, 176
searchInContributories, 176
setGroupFilter, 177
setSearchInBillsQuotes, 177
setSearchInCompanies, 177
setSearchInContributories, 177
setSearchInProjects, 177
setSearchInReferentLastname, 177
setText, 178
Models::Statistics, 185
getTurnoverBetweenDates, 185
Models::Unit, 193
getype, 193
operator!=, 194
operator==, 194
setType, 194
toString, 194
Unit, 193
Models::User, 196
getDataMap, 197
getNbDaysPerMonths, 198
getNbHoursPerDays, 198
getNoSiret, 198
getPdflatexPath, 198
getTitle, 198
getWorkspaceName, 198
getWorkspacePath, 198
hydrat, 199
operator!=, 199
operator==, 199
setNoSiret, 199
setPdflatexPath, 199
setTitle, 200
setWorkspaceName, 200
setWorkspacePath, 200
updateFolders, 200
User, 197
Mustache::Context, 71
canEval, 72
Context, 72
eval, 72
isFalse, 72
listCount, 72
partialResolver, 72
partialValue, 72
pop, 72
push, 73
stringValue, 73
Mustache::PartialFileLoader, 134
getPartial, 134
Mustache::PartialMap, 134
getPartial, 135
Mustache::PartialResolver, 135
getPartial, 136
Mustache::QtVariantContext, 164
canEval, 165
eval, 165
fn_t, 165
isFalse, 166
listCount, 166
pop, 166
push, 166
stringValue, 166
Mustache::Renderer, 172
error, 172
errorPartial, 172
errorPos, 172
render, 172
setTagMarkers, 173

Mustache::Tag, 186
Comment, 187
InvertedSectionStart, 187
Partial, 187
SectionEnd, 187
SectionStart, 187
SetDelimiter, 187
Type, 187

operator!=
Models::Billing, 29
Models::Contributory, 87
Models::People, 140
Models::Project, 147
Models::Unit, 194
Models::User, 199
operator<
Models::Billing, 30
Models::Project, 147
operator<<
Utils::Log, 127
operator==
Models::Billing, 30
Models::Contributory, 87
Models::People, 140
Models::Project, 147
Models::Unit, 194
Models::User, 199

paint
Gui::Widgets::Delegates::ComboBoxDelegate, 67
Gui::Widgets::Delegates::ProjectComboDelegate, 150
Gui::Widgets::Delegates::UnitComboDelegate, 195
Parameters, 133
DB_FILENAME, 133
VERSION, 134
Partial
Mustache::Tag, 187
partialResolver
Mustache::Context, 72
partialValue
Mustache::Context, 72
PdfGenerator
Generator::PdfGenerator, 136
PointersTest, 143
pop
Mustache::Context, 72
Mustache::QtVariantContext, 166
popupMessage
Exceptions::DbException, 112
Exceptions::FileException, 119
printInformations
Gui::Widgets::CustomerDataWidget, 102
Project
Models::Project, 145
ProjectComboDelegate
Gui::Widgets::Delegates::ProjectComboDelegate, 150
ProjectContributoriesTableModel
Gui::Widgets::WdgModels::ProjectContributoriesTableModel, 152
ProjectDatabaseTest, 159
ProjectModelTest, 160
ProjectsTableModel
Gui::Widgets::WdgModels::ProjectsTableModel, 161
ProjectsWidget
Gui::Widgets::ProjectsWidget, 164
push
Mustache::Context, 73
Mustache::QtVariantContext, 166

Rate
Models::Rate, 167
RateModelTest, 170
RateWidget
Gui::Widgets::RateWidget, 171
remove
Gui::Widgets::WdgModels::BillingsTableModel, 41
Gui::Widgets::WdgModels::ContributoriesTableModel, 81
Gui::Widgets::WdgModels::CustomersTableModel, 106
Gui::Widgets::WdgModels::ProjectContributoriesTableModel, 154
Gui::Widgets::WdgModels::ProjectsTableModel, 162
removeBilling
Databases::BillingDatabase, 36
removeBillingProject
Databases::BillingDatabase, 37
removeContributory
Databases::ContributoryDatabase, 91
removeCustomer
Databases::CustomerDatabase, 100
removeProject
Databases::ProjectDatabase, 159
render
Mustache::Renderer, 172
resizeEvent
Gui::MainWindow, 131
rootTree
Gui::MainWindow, 131
round
Utils::Double, 114
rowCount
Gui::Widgets::WdgModels::BillingsTableModel, 41
Gui::Widgets::WdgModels::ContributoriesTableModel, 81
Gui::Widgets::WdgModels::CustomersTableModel, 106
Gui::Widgets::WdgModels::ProjectContributoriesTableModel, 154
Gui::Widgets::WdgModels::ProjectsTableModel, 162

search

Gui::Docks::SearchDock, 180
Gui::MainWindow, 131
Gui::Widgets::searchWidget, 182
SearchDock
Gui::Docks::SearchDock, 179
searchInContributories
Models::Search, 176
searchTest, 180
searchWidget
Gui::Widgets::searchWidget, 181
SectionEnd
Mustache::Tag, 187
SectionStart
Mustache::Tag, 187
selectCustomer
Gui::Widgets::searchWidget, 182
setAddress
Models::People, 140
setAddressComplement
Models::People, 141
setAllIdContributories
Models::ContributoriesList, 78
setBeginDate
Models::Project, 148
setBtnValid
Gui::Widgets::CheckFields::CheckQLineEdit, 57
setCity
Models::People, 141
setCompany
Models::People, 141
setCopy
Gui::Dialogs::AddQuoteDialog, 24
setCost
Models::Project, 148
setCountry
Gui::Widgets::CheckFields::CheckPhone, 53
Gui::Widgets::CheckFields::CheckPostalCode, 55
Models::People, 141
setCustomer
Models::Project, 148
setDailyRate
Gui::Widgets::RateWidget, 171
Models::Project, 148
setData
Gui::Widgets::WdgModels::BillingsTableModel, 41
Gui::Widgets::WdgModels::ContributoriesTable↩
Model, 81
Gui::Widgets::WdgModels::CustomersTableModel,
106
Gui::Widgets::WdgModels::ProjectContributories↩
TableModel, 154
Gui::Widgets::WdgModels::ProjectsTableModel,
163
setDatabase
Databases::Database, 109
setDate
Models::Billing, 30
SetDelimiter
Mustache::Tag, 187
setDescription
Models::Billing, 30
Models::Contributory, 87
Models::Project, 148
setEditorData
Gui::Widgets::Delegates::ComboBoxDelegate, 67
Gui::Widgets::Delegates::DoubleSpinBoxDelegate,
116
Gui::Widgets::Delegates::TextareaDelegate, 191
setEmail
Models::People, 141
setEndDate
Models::Project, 148
setFax
Models::People, 141
setField
Gui::Widgets::Path::ChosePathWidget, 65
setFirstname
Models::People, 142
setGroupFilter
Models::Search, 177
setHourlyRate
Gui::Widgets::RateWidget, 171
Models::Contributory, 88
setId
Models::IModel, 123
setIdBilling
Models::ContributoriesList, 78
setIdCustomer
Gui::Widgets::searchWidget, 182
setImage
Gui::Dialogs::MessageBox, 133
setInsert
Models::ContributoriesList, 78
setIsBilling
Models::Billing, 30
setIsPaid
Models::Billing, 31
setLastname
Models::People, 142
setLocked
Gui::Widgets::Delegates::ProjectComboDelegate,
151
setLongDescription
Models::Contributory, 88
setMobilePhone
Models::People, 142
setModelData
Gui::Widgets::Delegates::ComboBoxDelegate, 67
Gui::Widgets::Delegates::DoubleSpinBoxDelegate,
116
Gui::Widgets::Delegates::TextareaDelegate, 191
setName
Models::Project, 149
Utils::ItemType, 125
setNoSiret
Models::User, 199

setNumber
Models::Billing, 31
setPdflatexPath
Models::User, 199
setPhone
Models::People, 142
setPostalCode
Models::People, 142
setProject
Models::Contributory, 88
setQuantity
Models::Contributory, 88
setQuoteIdNumber
Gui::Dialogs::AddQuoteDialog, 25
setSearchInBillsQuotes
Models::Search, 177
setSearchInCompanies
Models::Search, 177
setSearchInContributories
Models::Search, 177
setSearchInProjects
Models::Search, 177
setSearchInReferentLastname
Models::Search, 177
setTagMarkers
Mustache::Renderer, 173
setText
Gui::Dialogs::MessageBox, 133
Models::Search, 178
setTitle
Models::Billing, 31
Models::User, 200
setToRemoved
Models::IModel, 123
setType
Models::Unit, 194
Utils::ItemType, 125
setTypeFiles
Gui::Widgets::Path::ChoseFileWidget, 64
setUnit
Models::Contributory, 88
setWebsite
Models::People, 142
setWidgetDailyRateValue
Gui::Widgets::RateWidget, 171
setWidgetHourlyRateValue
Gui::Widgets::RateWidget, 172
setWorkspaceName
Models::User, 200
setWorkspacePath
Models::User, 200
StartedWindowsDialog
Gui::Dialogs::StartedWindowsDialog, 184
StringTest, 186
stringValue
CounterContext, 93
Mustache::Context, 73
Mustache::QtVariantContext, 166
TestAdder< T >, 187
TestMustache, 188
TestRunner, 188
testadder, 187
TexGenerator
Generator::TexGenerator, 189
textChanged
Gui::Docks::SearchDock, 180
TextareaDelegate
Gui::Widgets::Delegates::TextareaDelegate, 191
toString
Models::Unit, 194
treeLevel
Gui::MainWindow, 131
Type
Mustache::Tag, 187

Unit
Models::Unit, 193
UnitComboDelegate
Gui::Widgets::Delegates::UnitComboDelegate, 195
updateBtn
Gui::Widgets::ProjectsWidget, 164
updateCustomer
Databases::CustomerDatabase, 100
updateEditorGeometry
Gui::Widgets::Delegates::ComboBoxDelegate, 69
Gui::Widgets::Delegates::DoubleSpinBoxDelegate, 116
Gui::Widgets::Delegates::TextareaDelegate, 191
updateFolders
Models::User, 200
updateProject
Databases::ProjectDatabase, 159
updateRateProject
Databases::RateDatabase, 169
User
Models::User, 197
UserDataDialog
Gui::Dialogs::UserDataDialog, 203
UserDatabaseTest, 202
UserModelTest, 203
Utils::Directories, 114
makeDirectory, 114
Utils::Double, 114
round, 114
Utils::HierarchicalSystem, 119
addBillingToProject, 120
addProjectToCustomer, 120
getCustomers, 120
getProjects, 120
Utils::ItemType, 124
getModel, 125
getName, 125
getType, 125
ItemType, 124
setName, 125
setType, 125
Utils::Log, 126

instance, 126
operator<<, 127
write, 127
Utils::String, 185
firstLetterToUpper, 186
Utils::pointers, 143

VERSION
Parameters, 134
value
Databases::Database, 109

write
Utils::Log, 127