

# QTypingTest

0.0.1

Generated by Doxygen 1.8.11



# Contents

<b>1</b>	<b>Qtyping test</b>	<b>1</b>
<b>2</b>	<b>Namespace Index</b>	<b>3</b>
2.1	Namespace List . . . . .	3
<b>3</b>	<b>Hierarchical Index</b>	<b>5</b>
3.1	Class Hierarchy . . . . .	5
<b>4</b>	<b>Class Index</b>	<b>7</b>
4.1	Class List . . . . .	7
<b>5</b>	<b>File Index</b>	<b>9</b>
5.1	File List . . . . .	9
<b>6</b>	<b>Namespace Documentation</b>	<b>11</b>
6.1	factory Namespace Reference . . . . .	11
6.1.1	Function Documentation . . . . .	11
6.1.1.1	findClosestSpace(const QString &search, int indexStart) . . . . .	11
6.1.1.2	findExistingWords(QString authorizedLetters, QString filename, QString must↔ Contain="") . . . . .	11
6.1.1.3	generateFromLetters(QStringList letterList, int length=50) . . . . .	12
6.1.1.4	generateLearning(QStringList mainLetter, QStringList allLetters) . . . . .	12
6.1.1.5	generatePractice(QStringList letters, bool onlyRealWords=true) . . . . .	12
6.1.1.6	generateWords(QString authorizedLetters, QString language, QString main↔ Letters="") . . . . .	13
6.1.1.7	selectRandomElement(QStringList listString) . . . . .	13
6.1.1.8	splitText(QString toSplit, int numberOfSplit) . . . . .	13
6.2	file Namespace Reference . . . . .	14
6.2.1	Function Documentation . . . . .	14
6.2.1.1	getPadlockIcon(bool open) . . . . .	14
6.2.1.2	getStylesheet(QString stylesheetname) . . . . .	14
6.2.1.3	getWordsPath(QString lang) . . . . .	14
6.3	html Namespace Reference . . . . .	14
6.3.1	Function Documentation . . . . .	15

6.3.1.1	<code>addTags(QString &amp;strToTag, QString tags, QString attributes="")</code>	15
6.3.1.2	<code>addTagToChars(QString strToTag, QString tag, QString attributes="")</code>	15
6.3.1.3	<code>getAbsoluteCharPosition(QString string, int relativePos)</code>	15
6.3.1.4	<code>removeTag(QString &amp;strToClean, QString tag="([&gt;]*)")</code>	16
6.4	Ui Namespace Reference	16
<b>7</b>	<b>Class Documentation</b>	<b>17</b>
7.1	<code>date_exercice_</code> Struct Reference	17
7.1.1	Member Data Documentation	17
7.1.1.1	<code>dateResult</code>	17
7.1.1.2	<code>exercice</code>	17
7.2	<code>LearnPage</code> Class Reference	17
7.2.1	Constructor & Destructor Documentation	18
7.2.1.1	<code>LearnPage(QWidget *parent=0)</code>	18
7.2.1.2	<code>LearnPage(const LearnPage &amp;orig)</code>	18
7.2.1.3	<code>~LearnPage()</code>	18
7.2.2	Member Function Documentation	18
7.2.2.1	<code>endExercice</code>	18
7.2.2.2	<code>lauchExercice</code>	18
7.2.2.3	<code>resetExercice</code>	18
7.3	<code>Ui::MainWindow</code> Class Reference	18
7.4	<code>PracticePage</code> Class Reference	18
7.4.1	Constructor & Destructor Documentation	19
7.4.1.1	<code>PracticePage(QWidget *parent=0)</code>	19
7.4.1.2	<code>PracticePage(const PracticePage &amp;orig)</code>	19
7.4.1.3	<code>~PracticePage()</code>	19
7.4.2	Member Function Documentation	19
7.4.2.1	<code>saveExerciceResult</code>	19
7.4.2.2	<code>startAgainstTime</code>	19
7.5	<code>TestLearnDialog</code> Class Reference	19
7.6	<code>TestPracticeRace</code> Class Reference	20
7.7	<code>TestProgression</code> Class Reference	20
7.8	<code>TestTUser</code> Class Reference	20
7.9	<code>TestUtils</code> Class Reference	20
7.10	<code>TExercice</code> Class Reference	21
7.10.1	Detailed Description	21
7.10.2	Member Enumeration Documentation	21
7.10.2.1	<code>EXERCICE_TYPE</code>	21
7.10.3	Constructor & Destructor Documentation	21

7.10.3.1	<a href="#">TExercice(EXERCICE_TYPE exType, QStringList mainLetter, QStringList availableLetters)</a>	21
7.10.3.2	<a href="#">TExercice(EXERCICE_TYPE exType, bool allLetters)</a>	21
7.10.3.3	<a href="#">TExercice(const TExercice &amp;orig)</a>	21
7.10.3.4	<a href="#">~TExercice()</a>	21
7.10.4	<a href="#">Member Function Documentation</a>	22
7.10.4.1	<a href="#">buildExercice() const</a>	22
7.11	<a href="#">THomePage Class Reference</a>	22
7.11.1	<a href="#">Constructor &amp; Destructor Documentation</a>	22
7.11.1.1	<a href="#">THomePage(QWidget *parent=0)</a>	22
7.11.1.2	<a href="#">~THomePage()</a>	22
7.11.2	<a href="#">Member Function Documentation</a>	22
7.11.2.1	<a href="#">getUser()</a>	22
7.11.3	<a href="#">Member Data Documentation</a>	22
7.11.3.1	<a href="#">currentUser_</a>	22
7.12	<a href="#">TKeys Class Reference</a>	23
7.12.1	<a href="#">Constructor &amp; Destructor Documentation</a>	23
7.12.1.1	<a href="#">TKeys()</a>	23
7.12.1.2	<a href="#">~TKeys()</a>	23
7.12.2	<a href="#">Member Function Documentation</a>	23
7.12.2.1	<a href="#">isValidLetter(int letterCode)</a>	23
7.12.2.2	<a href="#">validLetters()</a>	23
7.13	<a href="#">TLabel Class Reference</a>	23
7.13.1	<a href="#">Constructor &amp; Destructor Documentation</a>	24
7.13.1.1	<a href="#">TLabel(QString text, QWidget *parent=0)</a>	24
7.13.1.2	<a href="#">~TLabel()</a>	24
7.13.2	<a href="#">Member Function Documentation</a>	24
7.13.2.1	<a href="#">eraseOverflow</a>	24
7.13.2.2	<a href="#">nextChar(bool currendIsCorrect)</a>	24
7.13.2.3	<a href="#">previousChar()</a>	24
7.13.2.4	<a href="#">setFirst()</a>	24
7.14	<a href="#">TPage Class Reference</a>	24
7.14.1	<a href="#">Constructor &amp; Destructor Documentation</a>	25
7.14.1.1	<a href="#">TPage(QWidget *parent=0)</a>	25
7.14.1.2	<a href="#">TPage(QString model, QWidget *parent)</a>	25
7.14.1.3	<a href="#">TPage(QString model)</a>	25
7.14.2	<a href="#">Member Function Documentation</a>	25
7.14.2.1	<a href="#">endedPage</a>	25
7.14.2.2	<a href="#">eraseAnswer</a>	25
7.14.2.3	<a href="#">getResult()</a>	25

7.14.2.4	<a href="#">startedPage</a>	25
7.14.2.5	<a href="#">text()</a>	26
7.14.2.6	<a href="#">typingAnswer</a>	26
7.14.2.7	<a href="#">update(QKeyEvent *ev)</a>	26
7.14.2.8	<a href="#">updateAsFirst()</a>	26
7.15	<a href="#">TPractical Class Reference</a>	26
7.15.1	<a href="#">Member Enumeration Documentation</a>	26
7.15.1.1	<a href="#">DIFFICULTY</a>	26
7.15.2	<a href="#">Constructor &amp; Destructor Documentation</a>	27
7.15.2.1	<a href="#">TPractical()</a>	27
7.15.2.2	<a href="#">TPractical(const TPractical &amp;orig)</a>	27
7.15.2.3	<a href="#">~TPractical()</a>	27
7.16	<a href="#">TPractice Class Reference</a>	27
7.16.1	<a href="#">Detailed Description</a>	27
7.16.2	<a href="#">Member Enumeration Documentation</a>	27
7.16.2.1	<a href="#">KEYBOARD_LAYOUT</a>	27
7.16.3	<a href="#">Constructor &amp; Destructor Documentation</a>	28
7.16.3.1	<a href="#">~TPractice()</a>	28
7.16.4	<a href="#">Member Function Documentation</a>	28
7.16.4.1	<a href="#">getAllLettersTo(int lastIndex=-1)</a>	28
7.16.4.2	<a href="#">getInstance()</a>	28
7.16.4.3	<a href="#">getLetterList()</a>	28
7.16.4.4	<a href="#">getLettersAt(int index)</a>	28
7.17	<a href="#">TPracticeRace Class Reference</a>	28
7.17.1	<a href="#">Constructor &amp; Destructor Documentation</a>	28
7.17.1.1	<a href="#">TPracticeRace(QWidget *parent=0)</a>	28
7.17.1.2	<a href="#">TPracticeRace(TExercise *generator, QWidget *parent=0)</a>	29
7.17.1.3	<a href="#">TPracticeRace(const TPracticeRace &amp;orig)</a>	29
7.17.1.4	<a href="#">~TPracticeRace()</a>	29
7.17.2	<a href="#">Member Function Documentation</a>	29
7.17.2.1	<a href="#">createPage</a>	29
7.17.2.2	<a href="#">exerciceGenerator() const</a>	29
7.17.2.3	<a href="#">updateClock</a>	29
7.18	<a href="#">TProgression Class Reference</a>	29
7.18.1	<a href="#">Detailed Description</a>	29
7.18.2	<a href="#">Constructor &amp; Destructor Documentation</a>	29
7.18.2.1	<a href="#">TProgression()</a>	29
7.18.2.2	<a href="#">TProgression(const TProgression &amp;orig)</a>	29
7.18.2.3	<a href="#">~TProgression()</a>	30
7.18.3	<a href="#">Member Function Documentation</a>	30

7.18.3.1	<a href="#">advanceExIndex()</a> . . . . .	30
7.18.3.2	<a href="#">getLastExericeIndex() const</a> . . . . .	30
7.18.3.3	<a href="#">setLastExericeIndex(int exIndex)</a> . . . . .	30
7.19	<a href="#">TResult Class Reference</a> . . . . .	30
7.19.1	<a href="#">Detailed Description</a> . . . . .	30
7.19.2	<a href="#">Constructor &amp; Destructor Documentation</a> . . . . .	30
7.19.2.1	<a href="#">TResult()</a> . . . . .	30
7.19.2.2	<a href="#">TResult(const TResult &amp;base)</a> . . . . .	30
7.19.2.3	<a href="#">~TResult()</a> . . . . .	30
7.19.3	<a href="#">Member Function Documentation</a> . . . . .	31
7.19.3.1	<a href="#">getCorrectKeysStrokes() const</a> . . . . .	31
7.19.3.2	<a href="#">getCorrectWords() const</a> . . . . .	31
7.19.3.3	<a href="#">getResume()</a> . . . . .	31
7.19.3.4	<a href="#">getTotalKeysStokres() const</a> . . . . .	31
7.19.3.5	<a href="#">getTotalWords() const</a> . . . . .	31
7.19.3.6	<a href="#">getWPM()</a> . . . . .	31
7.19.3.7	<a href="#">getWrongKeysStrokes() const</a> . . . . .	31
7.19.3.8	<a href="#">getWrongWords() const</a> . . . . .	31
7.19.3.9	<a href="#">incrCorrectKeystrokes(int incrementation)</a> . . . . .	31
7.19.3.10	<a href="#">incrWrongKeyStrokes(int incrementation)</a> . . . . .	31
7.19.3.11	<a href="#">operator+(const TResult &amp;otherRes)</a> . . . . .	31
7.19.3.12	<a href="#">operator+=(const TResult &amp;otherRes)</a> . . . . .	31
7.19.3.13	<a href="#">setCorrectKeysStrokes(int correctKeysStrokes)</a> . . . . .	31
7.19.3.14	<a href="#">setWrongKeysStrokes(int wrongKeysStrokes)</a> . . . . .	31
7.19.3.15	<a href="#">updateWPM(float deltaTime)</a> . . . . .	31
7.20	<a href="#">TStackPages Class Reference</a> . . . . .	31
7.20.1	<a href="#">Constructor &amp; Destructor Documentation</a> . . . . .	32
7.20.1.1	<a href="#">TStackPages(QWidget *parent=0)</a> . . . . .	32
7.20.1.2	<a href="#">TStackPages(const TStackPages &amp;orig)</a> . . . . .	32
7.20.1.3	<a href="#">TStackPages(const QString &amp;text, int numberOfPages=1, QWidget *parent=0)</a> . . . . .	32
7.20.1.4	<a href="#">~TStackPages()</a> . . . . .	32
7.20.2	<a href="#">Member Function Documentation</a> . . . . .	32
7.20.2.1	<a href="#">addPage(QString pageText, bool isFirst=false)</a> . . . . .	32
7.20.2.2	<a href="#">beginExerice</a> . . . . .	32
7.20.2.3	<a href="#">currentPage() const</a> . . . . .	32
7.20.2.4	<a href="#">exericeStarted</a> . . . . .	32
7.20.2.5	<a href="#">getText() const</a> . . . . .	32
7.20.2.6	<a href="#">keyPressed</a> . . . . .	32
7.20.2.7	<a href="#">nextPage</a> . . . . .	33
7.20.2.8	<a href="#">numberOfPages() const</a> . . . . .	33

7.20.2.9	pageEnded	33
7.20.2.10	setFollowingPage()	33
7.20.2.11	textFinished	33
7.21	TStats Class Reference	33
7.21.1	Constructor & Destructor Documentation	34
7.21.1.1	TStats()	34
7.21.1.2	TStats(const TStats &orig)	34
7.21.1.3	~TStats()	34
7.21.2	Member Function Documentation	34
7.21.2.1	add(const TStats &otherStat)	34
7.21.2.2	operator+(const TStats &otherStat)	34
7.21.2.3	operator+=(const TStats &otherStat)	34
7.22	TToolbar Class Reference	34
7.22.1	Constructor & Destructor Documentation	35
7.22.1.1	TToolbar(QWidget *parent=0)	35
7.22.1.2	TToolbar(const TToolbar &orig)	35
7.22.1.3	~TToolbar()	35
7.22.2	Member Function Documentation	35
7.22.2.1	incrementTimer	35
7.22.2.2	pauseClicked	35
7.22.2.3	pauseTimer	35
7.22.2.4	setLCDDisplayValue(double value)	35
7.22.2.5	setProgression(QString text)	35
7.23	TUser Class Reference	35
7.23.1	Constructor & Destructor Documentation	36
7.23.1.1	TUser(QString pseudo="")	36
7.23.1.2	~TUser()	36
7.23.2	Member Function Documentation	36
7.23.2.1	addResult(TExercice *exTyp, TResult *exRes)	36
7.23.2.2	getPracticeHistory() const	36
7.23.2.3	getProgression() const	36
7.23.2.4	getPseudo() const	36
7.23.2.5	getStatistics() const	36
7.23.2.6	oneMoreMistake(const QChar &mistaken)	36
7.23.2.7	setPseudo(QString pseudo)	36
7.24	TWindowLearn Class Reference	36
7.24.1	Constructor & Destructor Documentation	37
7.24.1.1	TWindowLearn(QString model, QWidget *parent=0)	37
7.24.1.2	TWindowLearn(QString model, int numberOfPages, QWidget *parent=0)	37
7.24.1.3	~TWindowLearn()	37



7.24.2	Member Function Documentation	37
7.24.2.1	updateClock()	37
7.25	TWindowTest Class Reference	37
7.25.1	Constructor & Destructor Documentation	39
7.25.1.1	TWindowTest(QWidget *parent=0)	39
7.25.1.2	TWindowTest(QString text, QWidget *parent=0)	39
7.25.1.3	TWindowTest(QString text, int numberOfPages, QWidget *parent=0)	39
7.25.1.4	~TWindowTest()	39
7.25.2	Member Function Documentation	39
7.25.2.1	beginExercise	39
7.25.2.2	closed	39
7.25.2.3	closeEvent(QCloseEvent *ev)	39
7.25.2.4	endOfExercise	39
7.25.2.5	exerciseFinished	39
7.25.2.6	exerciseResult()	39
7.25.2.7	getPageProgression()	40
7.25.2.8	keyPressEvent(QKeyEvent *ev)	40
7.25.2.9	moveEvent(QMoveEvent *ev)	40
7.25.2.10	pauseContinueExercise	40
7.25.2.11	saveResult	40
7.25.2.12	setTimers(int timeLimit)	40
7.25.2.13	setupShortcuts()	40
7.25.2.14	setupTimer()	41
7.25.2.15	setupWidget()	41
7.25.2.16	stackPages() const	41
7.25.2.17	toolBar() const	41
7.25.2.18	updateClock	41
7.25.2.19	updateToolbarProgression	41
7.25.3	Member Data Documentation	41
7.25.3.1	pages_	41
7.25.3.2	results_	41
7.25.3.3	topToolBar_	41
7.26	Ui_MainWindow Class Reference	41
7.26.1	Member Function Documentation	42
7.26.1.1	retranslateUi(QMainWindow *MainWindow)	42
7.26.1.2	setupUi(QMainWindow *MainWindow)	42
7.26.2	Member Data Documentation	42
7.26.2.1	actionChange_user	42
7.26.2.2	actionChange_user_2	42
7.26.2.3	actionHomepage	42

7.26.2.4	<a href="#">actionLanguage</a>	42
7.26.2.5	<a href="#">actionMap_keyboard</a>	42
7.26.2.6	<a href="#">actionOnline_competitoin</a>	42
7.26.2.7	<a href="#">actionOthers</a>	43
7.26.2.8	<a href="#">actionPreferences</a>	43
7.26.2.9	<a href="#">actionText_practice</a>	43
7.26.2.10	<a href="#">button_games</a>	43
7.26.2.11	<a href="#">button_home</a>	43
7.26.2.12	<a href="#">button_learn</a>	43
7.26.2.13	<a href="#">button_practice</a>	43
7.26.2.14	<a href="#">button_stats</a>	43
7.26.2.15	<a href="#">centralwidget</a>	43
7.26.2.16	<a href="#">gridLayout_2</a>	43
7.26.2.17	<a href="#">horizontalLayout_2</a>	43
7.26.2.18	<a href="#">label_3</a>	43
7.26.2.19	<a href="#">label_4</a>	43
7.26.2.20	<a href="#">menuAbout</a>	43
7.26.2.21	<a href="#">menubar</a>	43
7.26.2.22	<a href="#">menuOptions</a>	43
7.26.2.23	<a href="#">menuUser</a>	43
7.26.2.24	<a href="#">page_games</a>	43
7.26.2.25	<a href="#">page_home</a>	43
7.26.2.26	<a href="#">page_learn</a>	43
7.26.2.27	<a href="#">page_practice</a>	43
7.26.2.28	<a href="#">page_stats</a>	43
7.26.2.29	<a href="#">stack_main</a>	43
7.26.2.30	<a href="#">statusbar</a>	43
7.26.2.31	<a href="#">verticalLayout</a>	43
7.26.2.32	<a href="#">verticalLayoutWidget</a>	43
<b>8</b>	<b>File Documentation</b>	<b>45</b>
8.1	<a href="#">doxygen.txt File Reference</a>	45
8.2	<a href="#">etc/words/en.txt File Reference</a>	45
8.3	<a href="#">etc/words/fr.txt File Reference</a>	45
8.4	<a href="#">README.md File Reference</a>	45
8.5	<a href="#">src/Data/texercice.cpp File Reference</a>	45
8.6	<a href="#">src/Data/texercice.h File Reference</a>	45
8.7	<a href="#">src/Data/tkeys.cpp File Reference</a>	45
8.8	<a href="#">src/Data/tkeys.h File Reference</a>	46
8.9	<a href="#">src/Data/tpractical.cpp File Reference</a>	46

8.10	src/Data/tpractical.h File Reference	46
8.11	src/Data/tpractice.cpp File Reference	46
8.12	src/Data/tpractice.h File Reference	46
8.13	src/Data/tprogression.cpp File Reference	46
8.13.1	Function Documentation	47
8.13.1.1	operator<<(QDataStream &out, const TProgression &prog)	47
8.13.1.2	operator==(const TProgression &prog1, const TProgression &prog2)	47
8.13.1.3	operator>>(QDataStream &in, TProgression &prog)	47
8.14	src/Data/tprogression.h File Reference	47
8.14.1	Function Documentation	47
8.14.1.1	operator<<(QDataStream &out, const TProgression &prog)	47
8.14.1.2	operator==(const TProgression &prog1, const TProgression &prog2)	47
8.14.1.3	operator>>(QDataStream &in, TProgression &prog)	47
8.15	src/Data/tresult.cpp File Reference	47
8.16	src/Data/tresult.h File Reference	47
8.17	src/Data/tstats.cpp File Reference	47
8.18	src/Data/tstats.h File Reference	48
8.19	src/Data/tuser.cpp File Reference	48
8.19.1	Function Documentation	48
8.19.1.1	operator<<(QDataStream &out, const TUser &user)	48
8.19.1.2	operator>>(QDataStream &in, TUser &user)	48
8.20	src/Data/tuser.h File Reference	48
8.20.1	Function Documentation	49
8.20.1.1	operator<<(QDataStream &out, const TUser &user)	49
8.20.1.2	operator==(const TUser &user1, const TUser &user2)	49
8.20.1.3	operator>>(QDataStream &in, TUser &user)	49
8.21	src/main.cpp File Reference	49
8.21.1	Function Documentation	49
8.21.1.1	main(int argc, char *argv[])	49
8.22	src/QTypingTest/Dialogs/Exercices/tlabel.cpp File Reference	49
8.23	src/QTypingTest/Dialogs/Exercices/tlabel.h File Reference	49
8.24	src/QTypingTest/Dialogs/Exercices/tpage.cpp File Reference	49
8.25	src/QTypingTest/Dialogs/Exercices/tpage.h File Reference	50
8.26	src/QTypingTest/Dialogs/Exercices/tpracticerace.cpp File Reference	50
8.27	src/QTypingTest/Dialogs/Exercices/tpracticerace.h File Reference	50
8.28	src/QTypingTest/Dialogs/Exercices/tstackpages.cpp File Reference	50
8.29	src/QTypingTest/Dialogs/Exercices/tstackpages.h File Reference	51
8.30	src/QTypingTest/Dialogs/Exercices/ttoolbar.cpp File Reference	51
8.31	src/QTypingTest/Dialogs/Exercices/ttoolbar.h File Reference	51
8.32	src/QTypingTest/Dialogs/Exercices/twindowlearn.cpp File Reference	51

8.33	src/QTypingTest/Dialogs/Exercices/twindowlearn.h File Reference	51
8.34	src/QTypingTest/Dialogs/twindowtest.cpp File Reference	52
8.35	src/QTypingTest/Dialogs/twindowtest.h File Reference	52
8.36	src/QTypingTest/Learn/learnpage.cpp File Reference	52
8.37	src/QTypingTest/Learn/learnpage.h File Reference	53
8.37.1	Macro Definition Documentation	53
8.37.1.1	MAX_COL	53
8.38	src/QTypingTest/Practice/practicepage.cpp File Reference	53
8.39	src/QTypingTest/Practice/practicepage.h File Reference	53
8.40	src/QTypingTest/thomepage.cpp File Reference	54
8.41	src/QTypingTest/thomepage.h File Reference	54
8.42	src/ui/ui_homepage.h File Reference	54
8.43	src/Util/factory.cpp File Reference	55
8.43.1	Function Documentation	55
8.43.1.1	isValidWord(QString word, QString availableLetters)	55
8.43.1.2	randomNextSpace()	55
8.43.1.3	readFile(QString fileName)	55
8.43.1.4	wordsContains(QStringList words, QString contains)	55
8.43.2	Variable Documentation	55
8.43.2.1	numberOfWords	55
8.44	src/Util/factory.h File Reference	56
8.45	src/Util/filehelper.cpp File Reference	56
8.45.1	Variable Documentation	56
8.45.1.1	ETC_PATH	56
8.45.1.2	WORDS_EXT	56
8.45.1.3	WORDS_PATH	56
8.46	src/Util/filehelper.h File Reference	56
8.46.1	Variable Documentation	57
8.46.1.1	ETC_PATH	57
8.46.1.2	WORDS_EXT	57
8.46.1.3	WORDS_PATH	57
8.47	src/Util/htmlhelper.cpp File Reference	57
8.47.1	Function Documentation	57
8.47.1.1	surroundOfTags(QStringList tags, QString attributes)	57
8.48	src/Util/htmlhelper.h File Reference	58
8.49	tests/TestLearnDialog/testlearndialog.cpp File Reference	58
8.50	tests/TestPracticeRace/testpracticerace.cpp File Reference	58
8.51	tests/TestProgression/testprogression.cpp File Reference	59
8.52	tests/TestUser/testuser.cpp File Reference	59
8.53	tests/TestUtils/testutils.cpp File Reference	59

---

8.53.1 Variable Documentation . . . . .	60
8.53.1.1 allLetters . . . . .	60
<b>Index</b>	<b>61</b>



# Chapter 1

## Qtyping test

QTypingTest is a software developed with Qt. It teaches you how to type faster. You can also practice some random words just to improve your speed. Some statistics will be made to see your progression and you will be able to train on exercise composed of the letters you make the most mistake on.

*This project is realized for a university project*

### Learn

You can start from scratch and learn the position of the keys on you keyboard. Step by step.

### Practice

You have to type words. The difference with the learning exercises, is that there are only existing words. Not random-generated words.

### Text practice

Instead of typing series of random words, you have to type a real text. This exercise is the closest one to the reality.

### Previews





## Chapter 2

# Namespace Index

### 2.1 Namespace List

Here is a list of all namespaces with brief descriptions:

<a href="#">factory</a>	11
<a href="#">file</a>	14
<a href="#">html</a>	14
<a href="#">Ui</a>	16



## Chapter 3

# Hierarchical Index

### 3.1 Class Hierarchy

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

date_exercise_ . . . . .	17
QDialog	
TWindowTest . . . . .	37
TPracticeRace . . . . .	28
TWindowLearn . . . . .	36
QHash	
TStats . . . . .	33
QLabel	
TLabel . . . . .	23
QMainWindow	
THomePage . . . . .	22
QObject	
TestLearnDialog . . . . .	19
TestPracticeRace . . . . .	20
TestProgression . . . . .	20
TestTUser . . . . .	20
TestUtils . . . . .	20
QStackedWidget	
TStackPages . . . . .	31
QWidget	
LearnPage . . . . .	17
PracticePage . . . . .	18
TPage . . . . .	24
TToolbar . . . . .	34
TExercice . . . . .	21
TKeys . . . . .	23
TPractical . . . . .	26
TPractice . . . . .	27
TProgression . . . . .	29
TResult . . . . .	30
TUser . . . . .	35
Ui_MainWindow . . . . .	41
Ui::MainWindow . . . . .	18



## Chapter 4

# Class Index

### 4.1 Class List

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

<a href="#">date_exercice_</a>	17
<a href="#">LearnPage</a>	17
<a href="#">Ui::MainWindow</a>	18
<a href="#">PracticePage</a>	18
<a href="#">TestLearnDialog</a>	19
<a href="#">TestPracticeRace</a>	20
<a href="#">TestProgression</a>	20
<a href="#">TestTUser</a>	20
<a href="#">TestUtils</a>	20
<a href="#">TExercice</a>	21
<a href="#">THomePage</a>	22
<a href="#">TKeys</a>	23
<a href="#">TLabel</a>	23
<a href="#">TPage</a>	24
<a href="#">TPactical</a>	26
<a href="#">TPractice</a>	27
<a href="#">TPracticeRace</a>	28
<a href="#">TProgression</a>	29
<a href="#">TResult</a>	30
<a href="#">TStackPages</a>	31
<a href="#">TStats</a>	33
<a href="#">TToolbar</a>	34
<a href="#">TUser</a>	35
<a href="#">TWindowLearn</a>	36
<a href="#">TWindowTest</a>	37
<a href="#">Ui_MainWindow</a>	41



## Chapter 5

# File Index

### 5.1 File List

Here is a list of all files with brief descriptions:

src/main.cpp . . . . .	49
src/Data/texercice.cpp . . . . .	45
src/Data/texercice.h . . . . .	45
src/Data/tkeys.cpp . . . . .	45
src/Data/tkeys.h . . . . .	46
src/Data/tpractical.cpp . . . . .	46
src/Data/tpractical.h . . . . .	46
src/Data/tpractice.cpp . . . . .	46
src/Data/tpractice.h . . . . .	46
src/Data/tprogression.cpp . . . . .	46
src/Data/tprogression.h . . . . .	47
src/Data/tresult.cpp . . . . .	47
src/Data/tresult.h . . . . .	47
src/Data/tstats.cpp . . . . .	47
src/Data/tstats.h . . . . .	48
src/Data/tuser.cpp . . . . .	48
src/Data/tuser.h . . . . .	48
src/QTypingTest/thomepage.cpp . . . . .	54
src/QTypingTest/thomepage.h . . . . .	54
src/QTypingTest/Dialogs/twindowtest.cpp . . . . .	52
src/QTypingTest/Dialogs/twindowtest.h . . . . .	52
src/QTypingTest/Dialogs/Exercices/tlabel.cpp . . . . .	49
src/QTypingTest/Dialogs/Exercices/tlabel.h . . . . .	49
src/QTypingTest/Dialogs/Exercices/tpage.cpp . . . . .	49
src/QTypingTest/Dialogs/Exercices/tpage.h . . . . .	50
src/QTypingTest/Dialogs/Exercices/tpracticerace.cpp . . . . .	50
src/QTypingTest/Dialogs/Exercices/tpracticerace.h . . . . .	50
src/QTypingTest/Dialogs/Exercices/tstackpages.cpp . . . . .	50
src/QTypingTest/Dialogs/Exercices/tstackpages.h . . . . .	51
src/QTypingTest/Dialogs/Exercices/ttoolbar.cpp . . . . .	51
src/QTypingTest/Dialogs/Exercices/ttoolbar.h . . . . .	51
src/QTypingTest/Dialogs/Exercices/twindowlearn.cpp . . . . .	51
src/QTypingTest/Dialogs/Exercices/twindowlearn.h . . . . .	51
src/QTypingTest/Learn/learnpage.cpp . . . . .	52
src/QTypingTest/Learn/learnpage.h . . . . .	53
src/QTypingTest/Practice/practicepage.cpp . . . . .	53
src/QTypingTest/Practice/practicepage.h . . . . .	53
src/ui/ui_homepage.h . . . . .	54

src/Util/factory.cpp	55
src/Util/factory.h	56
src/Util/filehelper.cpp	56
src/Util/filehelper.h	56
src/Util/htmlhelper.cpp	57
src/Util/htmlhelper.h	58
tests/TestLearnDialog/testlearndialog.cpp	58
tests/TestPracticeRace/testpracticerace.cpp	58
tests/TestProgression/testprogression.cpp	59
tests/TestUser/testuser.cpp	59
tests/TestUtils/testutils.cpp	59



## Chapter 6

# Namespace Documentation

### 6.1 factory Namespace Reference

#### Functions

- QString [generateLearning](#) (QStringList mainLetter, QStringList [allLetters](#))
- QString [generatePractice](#) (QStringList letters, bool onlyRealWords=true)
- QString [generateFromLetters](#) (QStringList letterList, int length=50)
- QString [generateWords](#) (QString authorizedLetters, QString language, QString mainLetters="")
- QStringList [findExistingWords](#) (QString authorizedLetters, QString filename, QString mustContain="")
- QString [selectRandomElement](#) (QStringList listString)
- int [findClosestSpace](#) (const QString &search, int indexStart)
- QStringList [splitText](#) (QString toSplit, int numberOfSplit)

#### 6.1.1 Function Documentation

##### 6.1.1.1 int factory::findClosestSpace ( const QString & *search*, int *indexStart* )

Helper to find the closest space in a list of string to the be able to break the list at a space, instead of breaking in the middle of a string

#### Parameters

<i>search</i>	the list of string containing spaces
<i>indexStart</i>	the index where to start the reserach

#### Returns

the index of the closest space in the list from indexStart

##### 6.1.1.2 QStringList factory::findExistingWords ( QString *authorizedLetters*, QString *filename*, QString *mustContain* = " " )

Will look into a file (depending on the chosen langage) and will find all the words that can be created with the given letters

#### Parameters

<i>authorizedLetters</i>	the letters that can compose the words
<i>fileName</i>	the path to the file containing all the words

## Parameters

<i>mustContain</i>	a Qstring of letters that MUST be in the word
--------------------	---

## Returns

a list of word that can be created with the given letters

#### 6.1.1.3 QString factory::generateFromLetters ( QStringList *letterList*, int *length* = 50 )

Generate some random words with the available letters given in the list

## Parameters

<i>letterList</i>	a list of 'authorized' letters
<i>length</i>	length of the result string

## Returns

a string with random words proposed in the given string list in parameter

#### 6.1.1.4 QString factory::generateLearning ( QStringList *mainLetter*, QStringList *allLetters* )

This function generate the practice exercise from the given letters. The result is divided into three parts :

- The first part is the letters that the user must learn, and only these
- The second part contains all the letters the user learned before so that he can train on all the letters
- The thirs parts contains the words that can be created with the letters the user learned before

NOTE : if no words can be made with the available letters, it will be replaced by a set of random letters

## Parameters

<i>mainLetter</i>	the letter that the user is going to learn
<i>allLetters</i>	all the available letters for the exercise

## Returns

a random-generated list of words

#### 6.1.1.5 QString factory::generatePractice ( QStringList *letters*, bool *onlyRealWords* = true )

Generates random words from the given list of letters The string generated is a single big text of words If onlyreal-words is true, the text will contains only `real` words so the user can train on existing words otherwise, it will also contains some random-generated words The size of the produced string is made to be contained in only ONE page of an exercise.

TODO : add a parameter to choose the size of the text ? (Azarias)

## Parameters

<i>letters</i>	the available letters
<i>onlyRealWords</i>	if it must generates only words existing in a language, or if it can generate random words

## Returns

the generated text

6.1.1.6 QString factory::generateWords ( QString *authorizedLetters*, QString *language*, QString *mainLetters* = " " )

Generate existing words with the available letters TODO : add a language parameter

## Parameters

<i>authorizedLetters</i>	all the available chars to build a word
<i>language</i>	the language of the file to find
<i>mainLetters</i>	an optional QString. If existing, all the words must contain at least one letter of this string

## Returns

a string with `numberOfWord` words separated by space.

6.1.1.7 QString factory::selectRandomElement ( QStringList *listString* )

Select one random element in the availables elements of the list Helper to create exercices

## Parameters

<i>listString</i>	a list of elements
-------------------	--------------------

## Returns

a random-selected element in the list

6.1.1.8 QStringList factory::splitText ( QString *toSplit*, int *numberOfSplit* )

Will split the text into differents parts to be able to display the given original string into differents line with approximately the same length The split is made at the closes space found, to avoid beaking a word and split at a space instead

## Parameters

<i>text</i>	the text to split
<i>the</i>	number of string to put in the list

## Returns

the text splitted

## 6.2 file Namespace Reference

### Functions

- QString [getWordsPath](#) (QString lang)
- QString [getPadlockIcon](#) (bool open)
- QString [getStylesheet](#) (QString stylesheetname)

### 6.2.1 Function Documentation

#### 6.2.1.1 QString file::getPadlockIcon ( bool *open* )

Helper to get the icons of the padlocks

##### Parameters

<i>open</i>	if the padlock is open or not !
-------------	---------------------------------

##### Returns

the relative path to the image of the corresponding padlock

#### 6.2.1.2 QString file::getStylesheet ( QString *stylesheetname* )

Return the path of the style file

##### Parameters

<i>stylesheetname</i>	the name of the file with the styl
-----------------------	------------------------------------

##### Returns

the relative path to the .qss file containing the style

#### 6.2.1.3 QString file::getWordsPath ( QString *lang* )

Helper to find the most common word of a given language

##### Parameters

<i>language</i>	the language of the file where the words are stored
-----------------	---

##### Returns

the RELATIVE path to the file containing the most common words of the given language

## 6.3 html Namespace Reference

## Functions

- QString [addTagToChars](#) (QString strToTag, QString tag, QString attributes="")
- QString [removeTag](#) (QString &strToClean, QString tag="([>]\*)")
- QString [addTags](#) (QString &strToTag, QString tags, QString attributes="")
- int [getAbsoluteCharPosition](#) (QString string, int relativePos)

### 6.3.1 Function Documentation

#### 6.3.1.1 QString html::addTags ( QString & *strToTag*, QString *tags*, QString *attributes* = " " )

Surround the QChar at position given in parameter by all the tags given in parameter. The tags must be separated by a comma. For example "b,u" will add the tags "b" then "u". It is possible to add some attributes into the FIRST surrounding tag The function will also change the string passed in parameter

##### Parameters

<i>strToTag</i>	the QString to be tagged
<i>tags</i>	the tags comma-separated to add to the character (e.g. "b,u")
<i>attributes</i>	some optional attributes to add to around the string

##### Returns

the String with the char tagged

#### 6.3.1.2 QString html::addTagToChars ( QString *strToTag*, QString *tag*, QString *attributes* = " " )

Add a tag around ALL the characters of a QString

##### Parameters

<i>strToTag</i>	the str where to add surround every char by a tag
<i>tag</i>	the tag to surround with

##### Returns

the QString with all the chars surrounded by the tag

#### 6.3.1.3 int html::getAbsoluteCharPosition ( QString *string*, int *relativePos* )

Helper to get the exact position of a char, without taking in consideration the HTML tags that makes the relative position falsy For example int the string "<p>p</p>atrick" the position of the first char is 3 (p) and not 0 (<>)

##### Parameters

<i>string</i>	the string with some eventual html tags
<i>relativePos</i>	the position of the searched char

**Returns**

the absolute position of the researched char, return -1 if the pos was not found (relativePos too big)

**6.3.1.4 QString html::removeTag ( QString & *strToClean*, QString *tag* = " ( [ ^ > ] \* ) " )**

Remove every tag in the QString. If no tag is specified, all will be removed, else, only the tag passed in parameter.

**Parameters**

<i>strToClean</i>	the QString to clean
<i>tag</i>	the tag to remove. If empty, all tags found will be removed

**Returns**

the cleaned QString

## 6.4 Ui Namespace Reference

**Classes**

- class [MainWindow](#)

## Chapter 7

# Class Documentation

### 7.1 date\_exercise\_ Struct Reference

```
#include <tuser.h>
```

#### Public Attributes

- QDateTime [dateResult](#)
- [TExercise](#) \* [exercise](#)

#### 7.1.1 Member Data Documentation

7.1.1.1 QDateTime [date\\_exercise\\_::dateResult](#)

7.1.1.2 [TExercise](#)\* [date\\_exercise\\_::exercise](#)

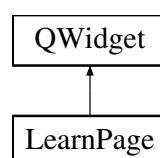
The documentation for this struct was generated from the following file:

- [src/Data/tuser.h](#)

### 7.2 LearnPage Class Reference

```
#include <learnpage.h>
```

Inheritance diagram for LearnPage:



#### Public Slots

- void [lauchExercise](#) ()
- void [endExercise](#) ([TResult](#) \*exerciseResult, QTime timeEx)
- void [resetExercise](#) ()

## Public Member Functions

- [LearnPage](#) (QWidget \*parent=0)
- [LearnPage](#) (const [LearnPage](#) &orig)
- virtual [~LearnPage](#) ()

### 7.2.1 Constructor & Destructor Documentation

7.2.1.1 [LearnPage::LearnPage](#) ( QWidget \* *parent* = 0 )

7.2.1.2 [LearnPage::LearnPage](#) ( const [LearnPage](#) & *orig* )

7.2.1.3 virtual [LearnPage::~~LearnPage](#) ( ) [inline],[virtual]

### 7.2.2 Member Function Documentation

7.2.2.1 void [LearnPage::endExercise](#) ( TResult \* *exerciseResult*, QTime *timeEx* ) [slot]

Called when an exercise is finished

7.2.2.2 void [LearnPage::lauchExercise](#) ( ) [slot]

7.2.2.3 void [LearnPage::resetExercise](#) ( ) [slot]

Called when an exercise is finished or when the exerice dialog is closed to reset all the necessary values

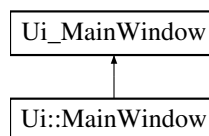
The documentation for this class was generated from the following files:

- [src/QTypingTest/Learn/learnpage.h](#)
- [src/QTypingTest/Learn/learnpage.cpp](#)

## 7.3 Ui::MainWindow Class Reference

```
#include <ui_homepage.h>
```

Inheritance diagram for Ui::MainWindow:



### Additional Inherited Members

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

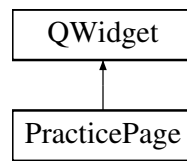
- [src/ui/ui\\_homepage.h](#)

## 7.4 PracticePage Class Reference

```
#include <practicepage.h>
```



Inheritance diagram for PracticePage:



## Public Slots

- void [startAgainstTime](#) ()
- void [saveExerciseResult](#) (TResult \*res, QTime time)

## Public Member Functions

- [PracticePage](#) (QWidget \*parent=0)
- [PracticePage](#) (const [PracticePage](#) &orig)
- virtual [~PracticePage](#) ()

### 7.4.1 Constructor & Destructor Documentation

7.4.1.1 `PracticePage::PracticePage ( QWidget * parent = 0 )`

7.4.1.2 `PracticePage::PracticePage ( const PracticePage & orig )`

7.4.1.3 `virtual PracticePage::~~PracticePage ( )` [inline],[virtual]

### 7.4.2 Member Function Documentation

7.4.2.1 `void PracticePage::saveExerciseResult ( TResult * res, QTime time )` [slot]

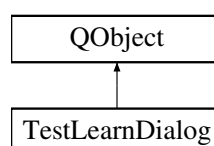
7.4.2.2 `void PracticePage::startAgainstTime ( )` [slot]

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

- src/QTypingTest/Practice/[practicepage.h](#)
- src/QTypingTest/Practice/[practicepage.cpp](#)

## 7.5 TestLearnDialog Class Reference

Inheritance diagram for TestLearnDialog:

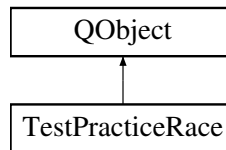


The documentation for this class was generated from the following file:

- tests/TestLearnDialog/[testlearndialog.cpp](#)

## 7.6 TestPracticeRace Class Reference

Inheritance diagram for TestPracticeRace:

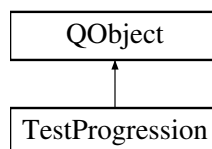


The documentation for this class was generated from the following file:

- tests/TestPracticeRace/[testpracticrace.cpp](#)

## 7.7 TestProgression Class Reference

Inheritance diagram for TestProgression:

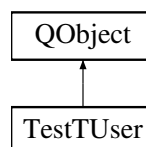


The documentation for this class was generated from the following file:

- tests/TestProgression/[testprogression.cpp](#)

## 7.8 TestTUser Class Reference

Inheritance diagram for TestTUser:

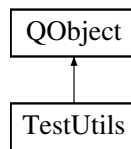


The documentation for this class was generated from the following file:

- tests/TestUser/[testuser.cpp](#)

## 7.9 TestUtils Class Reference

Inheritance diagram for TestUtils:



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

- tests/TestUtils/[testutils.cpp](#)

## 7.10 TExercise Class Reference

```
#include <texercise.h>
```

### Public Types

- enum [EXERCICE\\_TYPE](#) { [LEARNING](#), [PRACTICING](#), [TEXT\\_PRACTICING](#) }

### Public Member Functions

- [TExercise](#) ([EXERCICE\\_TYPE](#) exType, QStringList mainLetter, QStringList availableLetters)
- [TExercise](#) ([EXERCICE\\_TYPE](#) exType, bool [allLetters](#))
- [TExercise](#) (const [TExercise](#) &orig)
- QString [buildExercise](#) () const
- virtual [~TExercise](#) ()

### 7.10.1 Detailed Description

A model class to define the type of exercise the user is about to do There are differents types of exercices and depending of these, differents parameters

### 7.10.2 Member Enumeration Documentation

#### 7.10.2.1 enum TExercise::EXERCICE\_TYPE

Enumerator

***LEARNING***

***PRACTICING***

***TEXT\_PRACTICING***

### 7.10.3 Constructor & Destructor Documentation

7.10.3.1 TExercise::TExercise ( [EXERCICE\\_TYPE](#) exType, QStringList mainLetter, QStringList availableLetters )

7.10.3.2 TExercise::TExercise ( [EXERCICE\\_TYPE](#) exType, bool allLetters )

7.10.3.3 TExercise::TExercise ( const TExercise & orig )

7.10.3.4 virtual TExercise::~~TExercise ( ) [inline],[virtual]

## 7.10.4 Member Function Documentation

### 7.10.4.1 QString TExercise::buildExercise ( ) const

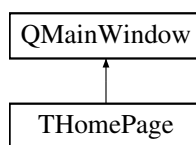
The documentation for this class was generated from the following files:

- src/Data/[texercise.h](#)
- src/Data/[texercise.cpp](#)

## 7.11 THomePage Class Reference

```
#include <thomepage.h>
```

Inheritance diagram for THomePage:



### Public Member Functions

- [THomePage](#) (QWidget \*parent=0)
- virtual [~THomePage](#) ()

### Static Public Member Functions

- static [TUser](#) \* [getUser](#) ()

### Static Public Attributes

- static [TUser](#) \* [currentUser\\_](#)

## 7.11.1 Constructor & Destructor Documentation

### 7.11.1.1 THomePage::THomePage ( QWidget \* parent = 0 )

### 7.11.1.2 THomePage::~~THomePage ( ) [virtual]

## 7.11.2 Member Function Documentation

### 7.11.2.1 static TUser\* THomePage::getUser ( ) [inline],[static]

## 7.11.3 Member Data Documentation

### 7.11.3.1 TUser \* THomePage::currentUser\_ [static]

Initiate the static vars

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

- src/QTypingTest/[thomepage.h](#)
- src/QTypingTest/[thomepage.cpp](#)

## 7.12 TKeys Class Reference

```
#include <tkeys.h>
```

### Public Member Functions

- [TKeys](#) ()
- virtual [~TKeys](#) ()

### Static Public Member Functions

- static QList< int > [validLetters](#) ()
- static bool [isValidLetter](#) (int letterCode)

### 7.12.1 Constructor & Destructor Documentation

7.12.1.1 [TKeys::TKeys](#) ( )

7.12.1.2 [virtual TKeys::~~TKeys](#) ( ) `[inline]`, `[virtual]`

### 7.12.2 Member Function Documentation

7.12.2.1 [bool TKeys::isValidLetter](#) ( int *letterCode* ) `[static]`

7.12.2.2 [QList< int > TKeys::validLetters](#) ( ) `[static]`

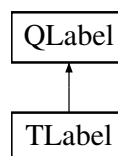
The documentation for this class was generated from the following files:

- [src/Data/tkeys.h](#)
- [src/Data/tkeys.cpp](#)

## 7.13 TLabel Class Reference

```
#include <tlabel.h>
```

Inheritance diagram for TLabel:



### Signals

- void [eraseOverflow](#) ()

### Public Member Functions

- [TLabel](#) (QString text, QWidget \*parent=0)
- virtual [~TLabel](#) ()

- bool [nextChar](#) (bool *currentIsCorrect*)
- bool [previousChar](#) ()
- void [setFirst](#) ()

### 7.13.1 Constructor & Destructor Documentation

7.13.1.1 `TLabel::TLabel ( QString text, QWidget * parent = 0 )`

7.13.1.2 `TLabel::~~TLabel ( )` [virtual]

### 7.13.2 Member Function Documentation

7.13.2.1 `void TLabel::eraseOverflow ( )` [signal]

A signal triggered when the user tries to erase whenever he's at the start of the line

7.13.2.2 `bool TLabel::nextChar ( bool currentIsCorrect )`

Update the current char color depending on the answer given by the user the boolean is here to say whether the answer was right or not

#### Parameters

<i>currentIsCorrect</i>	answer given by the user for the current char is correct
-------------------------	--

#### Returns

if there is another char after the current one

7.13.2.3 `bool TLabel::previousChar ( )`

Rollback the display and set the current label to the previous one in case of rollback from the user

#### Returns

if there are some chars before the current one

7.13.2.4 `void TLabel::setFirst ( )` [inline]

Function to update the display and show that the line is ready for use

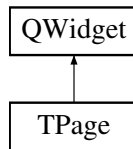
The documentation for this class was generated from the following files:

- [src/QTypingTest/Dialogs/Exercices/tlabel.h](#)
- [src/QTypingTest/Dialogs/Exercices/tlabel.cpp](#)

## 7.14 TPage Class Reference

```
#include <tpage.h>
```

Inheritance diagram for TPage:



## Public Slots

- void `typingAnswer` (QString answer)
- void `eraseAnswer` ()

## Signals

- void `startedPage` ()
- void `endedPage` (TResult \*)

## Public Member Functions

- `TPage` (QWidget \*parent=0)
- `TPage` (QString model, QWidget \*parent)
- `TPage` (QString model)
- void `update` (QKeyEvent \*ev)
- void `updateAsFirst` ()
- TResult \* `getResult` ()
- QString `text` ()

### 7.14.1 Constructor & Destructor Documentation

7.14.1.1 `TPage::TPage ( QWidget * parent = 0 )` [explicit]

7.14.1.2 `TPage::TPage ( QString model, QWidget * parent )`

7.14.1.3 `TPage::TPage ( QString model )`

### 7.14.2 Member Function Documentation

7.14.2.1 `void TPage::endedPage ( TResult * )` [signal]

Signal triggered when there is no more text to copy on the page

7.14.2.2 `void TPage::eraseAnswer ( )` [slot]

7.14.2.3 `TResult* TPage::getResult ( )` [inline]

7.14.2.4 `void TPage::startedPage ( )` [signal]

A Signal triggered when the user typed the first character of the line

7.14.2.5 `QString TPage::text ( )` `[inline]`

7.14.2.6 `void TPage::typingAnswer ( QString answer )` `[slot]`

7.14.2.7 `void TPage::update ( QKeyEvent * ev )`

Called by the parent dialog/Window whenever a key is pressed to update the view and show the user where he is in the answer

#### Parameters

<i>ev</i>	the key event that the user triggered
-----------	---------------------------------------

7.14.2.8 `void TPage::updateAsFirst ( )` `[inline]`

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

- [src/QTypingTest/Dialogs/Exercices/tpage.h](#)
- [src/QTypingTest/Dialogs/Exercices/tpage.cpp](#)

## 7.15 TPractical Class Reference

```
#include <tpactical.h>
```

### Public Types

- enum `DIFFICULTY` {  
`BASIC`, `EASY`, `NORMAL`, `MEDIUM`,  
`HARD`, `CHUCK_NORRIS` }

### Public Member Functions

- `TPractical ( )`
- `TPractical (const TPractical &orig)`
- virtual `~TPractical ( )`

## 7.15.1 Member Enumeration Documentation

7.15.1.1 enum `TPractical::DIFFICULTY`

#### Enumerator

***BASIC***

***EASY***

***NORMAL***

***MEDIUM***

***HARD***

***CHUCK\_NORRIS***



## 7.15.2 Constructor & Destructor Documentation

7.15.2.1 TPractical::TPractical ( )

7.15.2.2 TPractical::TPractical ( const TPractical & *orig* )

7.15.2.3 TPractical::~~TPractical ( ) [virtual]

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

- src/Data/[tpractical.h](#)
- src/Data/[tpractical.cpp](#)

## 7.16 TPractice Class Reference

```
#include <tpractice.h>
```

### Public Types

- enum [KEYBOARD\\_LAYOUT](#) { [QWERTY](#), [AZERTY](#), [QWERTZ](#) }

### Public Member Functions

- QStringList [getLetterList](#) ()
- QStringList [getAllLettersTo](#) (int lastIndex=-1)
- QStringList [getLettersAt](#) (int index)
- virtual [~TPractice](#) ()

### Static Public Member Functions

- static [TPractice](#) & [getInstance](#) ()

## 7.16.1 Detailed Description

This class is here to have a way to create the exercices on a certain order based on the keyboard layout

## 7.16.2 Member Enumeration Documentation

7.16.2.1 enum TPractice::KEYBOARD\_LAYOUT

Enumerator

**[QWERTY](#)**

**[AZERTY](#)**

**[QWERTZ](#)**

### 7.16.3 Constructor & Destructor Documentation

7.16.3.1 `virtual TPractice::~~TPractice ( ) [inline],[virtual]`

### 7.16.4 Member Function Documentation

7.16.4.1 `QStringList TPractice::getAllLettersTo ( int lastIndex = -1 )`

7.16.4.2 `static TPractice& TPractice::getInstance ( ) [inline],[static]`

7.16.4.3 `QStringList TPractice::getLetterList ( ) [inline]`

7.16.4.4 `QStringList TPractice::getLettersAt ( int index )`

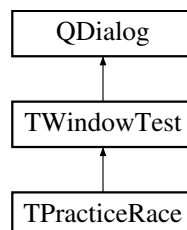
The documentation for this class was generated from the following files:

- [src/Data/tppractice.h](#)
- [src/Data/tppractice.cpp](#)

## 7.17 TPracticeRace Class Reference

```
#include <tppracticrace.h>
```

Inheritance diagram for TPracticeRace:



### Public Slots

- void [updateClock](#) ( )
- void [createPage](#) ( [TResult](#) \*prev=0 )

### Public Member Functions

- [TPracticeRace](#) ( [QWidget](#) \*parent=0 )
- [TPracticeRace](#) ( [TExercice](#) \*generator, [QWidget](#) \*parent=0 )
- [TPracticeRace](#) ( const [TPracticeRace](#) &orig )
- virtual [~TPracticeRace](#) ( )
- [TExercice](#) [exerciceGenerator](#) ( ) const

### Additional Inherited Members

#### 7.17.1 Constructor & Destructor Documentation

7.17.1.1 `TPracticeRace::TPracticeRace ( QWidget * parent = 0 )`

7.17.1.2 TPracticeRace::TPracticeRace ( TExercice \* *generator*, QWidget \* *parent* = 0 )

7.17.1.3 TPracticeRace::TPracticeRace ( const TPracticeRace & *orig* )

7.17.1.4 virtual TPracticeRace::~~TPracticeRace ( ) [inline], [virtual]

## 7.17.2 Member Function Documentation

7.17.2.1 void TPracticeRace::createPage ( TResult \* *prev* = 0 ) [slot]

Generate a new page for the exercise

Parameters

<i>prevPageRes</i>	the result of the page that was just finished
--------------------	---

7.17.2.2 TExercice TPracticeRace::exerciceGenerator ( ) const [inline]

7.17.2.3 void TPracticeRace::updateClock ( ) [slot]

decrement the time on the counter

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

- src/QTypingTest/Dialogs/Exercices/[tpracticrace.h](#)
- src/QTypingTest/Dialogs/Exercices/[tpracticrace.cpp](#)

## 7.18 TProgression Class Reference

```
#include <tprogression.h>
```

### Public Member Functions

- [TProgression](#) ( )
- [TProgression](#) (const [TProgression](#) &*orig*)
- virtual [~TProgression](#) ( )
- int [getLastExerciceIndex](#) ( ) const
- void [setLastExerciceIndex](#) (int *exIndex*)
- void [avdvanceExIndex](#) ( )

### 7.18.1 Detailed Description

A class to define the progression of a user to know what's the last letters he learned the difficulty of the exercices

TODO : add progression for the practice exercices

### 7.18.2 Constructor & Destructor Documentation

7.18.2.1 TProgression::TProgression ( )

7.18.2.2 TProgression::TProgression ( const TProgression & *orig* )

7.18.2.3 `TProgression::~~TProgression ( )` [virtual]

### 7.18.3 Member Function Documentation

7.18.3.1 `void TProgression::advanceExIndex ( )` [inline]

7.18.3.2 `int TProgression::getLastExerciseIndex ( ) const` [inline]

7.18.3.3 `void TProgression::setLastExerciseIndex ( int exIndex )` [inline]

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

- [src/Data/tprogression.h](#)
- [src/Data/tprogression.cpp](#)

## 7.19 TResult Class Reference

```
#include <tresult.h>
```

### Public Member Functions

- [TResult \( \)](#)
- [TResult \(const TResult &base\)](#)
- [virtual ~TResult \( \)](#)
- [int getCorrectKeysStrokes \( \) const](#)
- [int getWrongKeysStrokes \( \) const](#)
- [int getCorrectWords \( \) const](#)
- [int getWrongWords \( \) const](#)
- [int getWPM \( \)](#)
- [void setCorrectKeysStrokes \(int correctKeysStrokes\)](#)
- [void setWrongKeysStrokes \(int wrongKeysStrokes\)](#)
- [void incrWrongKeyStrokes \(int incrementation\)](#)
- [void incrCorrectKeystrokes \(int incrementation\)](#)
- [QString getResume \( \)](#)
- [int updateWPM \(float deltaTime\)](#)
- [int getTotalWords \( \) const](#)
- [int getTotalKeysStrokes \( \) const](#)
- [TResult \\* operator+ \(const TResult &otherRes\)](#)
- [TResult & operator+= \(const TResult &otherRes\)](#)

### 7.19.1 Detailed Description

This class is here to calculate the results of an exercise this must be used to calculate the speed, and the accuracy

### 7.19.2 Constructor & Destructor Documentation

7.19.2.1 `TResult::TResult ( )`

7.19.2.2 `TResult::TResult ( const TResult & base )`

7.19.2.3 `TResult::~~TResult ( )` [virtual]

### 7.19.3 Member Function Documentation

7.19.3.1 `int TResult::getCorrectKeysStrokes ( ) const` `[inline]`

7.19.3.2 `int TResult::getCorrectWords ( ) const` `[inline]`

7.19.3.3 `QString TResult::getResume ( )`

Return a QString with all the values of the object Helper to show results at the end of an exercise or when watching the history of exercises

#### Returns

the resume of all the data

7.19.3.4 `int TResult::getTotalKeysStrokes ( ) const`

7.19.3.5 `int TResult::getTotalWords ( ) const`

7.19.3.6 `int TResult::getWPM ( )` `[inline]`

7.19.3.7 `int TResult::getWrongKeysStrokes ( ) const` `[inline]`

7.19.3.8 `int TResult::getWrongWords ( ) const` `[inline]`

7.19.3.9 `void TResult::incrCorrectKeystrokes ( int incrementation )` `[inline]`

7.19.3.10 `void TResult::incrWrongKeyStrokes ( int incrementation )` `[inline]`

7.19.3.11 `TResult * TResult::operator+ ( const TResult & otherRes )`

7.19.3.12 `TResult & TResult::operator+= ( const TResult & otherRes )`

7.19.3.13 `void TResult::setCorrectKeysStrokes ( int correctKeysStrokes )` `[inline]`

7.19.3.14 `void TResult::setWrongKeysStrokes ( int wrongKeysStrokes )` `[inline]`

7.19.3.15 `int TResult::updateWPM ( float deltaTime )`

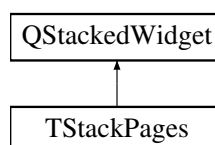
The documentation for this class was generated from the following files:

- [src/Data/tresult.h](#)
- [src/Data/tresult.cpp](#)

## 7.20 TStackPages Class Reference

```
#include <tstackpages.h>
```

Inheritance diagram for TStackPages:



## Public Slots

- void `nextPage` (`TResult` \*previousScore)
- void `beginExercise` ()
- void `keyPressed` (`QKeyEvent` \*ev)

## Signals

- void `pageEnded` (`TResult` \*pageResult)
- void `textFinished` ()
- void `exerciceStarted` ()

## Public Member Functions

- `TStackPages` (`QWidget` \*parent=0)
- `TStackPages` (const `TStackPages` &orig)
- `TStackPages` (const `QString` &text, int `numberOfPages`=1, `QWidget` \*parent=0)
- `QString` `getText` () const
- int `numberOfPages` () const
- `TPage` \* `currentPage` () const
- void `addPage` (`QString` pageText, bool isFirst=false)
- bool `setFollowingPage` ()
- virtual `~TStackPages` ()

### 7.20.1 Constructor & Destructor Documentation

7.20.1.1 `TStackPages::TStackPages ( QWidget * parent = 0 )`

7.20.1.2 `TStackPages::TStackPages ( const TStackPages & orig )`

7.20.1.3 `TStackPages::TStackPages ( const QString & text, int numberOfPages = 1, QWidget * parent = 0 )`

7.20.1.4 `virtual TStackPages::~~TStackPages ( )` `[inline]`, `[virtual]`

### 7.20.2 Member Function Documentation

7.20.2.1 `void TStackPages::addPage ( QString pageText, bool isFirst = false )`

7.20.2.2 `void TStackPages::beginExercise ( )` `[slot]`

Called whenever the user typed the first char on his keyboard This permits to launch a timer.

7.20.2.3 `TPage* TStackPages::currentPage ( ) const` `[inline]`

7.20.2.4 `void TStackPages::exerciceStarted ( )` `[signal]`

Called when the user typed the first letter of the exercice wether the letter is right

7.20.2.5 `QString TStackPages::getText ( ) const` `[inline]`

7.20.2.6 `void TStackPages::keyPressed ( QKeyEvent * ev )` `[slot]`

Called whenever the user type on the keybaord and the game is not paused

## Parameters

<code>ev</code>	the key event triggered by the user
-----------------	-------------------------------------

7.20.2.7 void TStackPages::nextPage ( TResult \* *previousScore* ) [slot]

To call whenever a page is finished and the next one must begin !

## Parameters

<i>previousScore</i>	the score of the line that was finished
----------------------	---

7.20.2.8 int TStackPages::numberOfPages ( ) const [inline]

7.20.2.9 void TStackPages::pageEnded ( TResult \* *pageResult* ) [signal]

Called whenever the current page is terminated

7.20.2.10 bool TStackPages::setFollowingPage ( )

7.20.2.11 void TStackPages::textFinished ( ) [signal]

Triggerred wherer all the given text at the beginning was copied

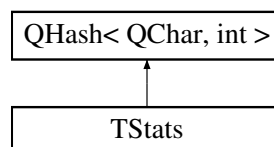
The documentation for this class was generated from the following files:

- src/QTypingTest/Dialogs/Exercices/[tstackpages.h](#)
- src/QTypingTest/Dialogs/Exercices/[tstackpages.cpp](#)

## 7.21 TStats Class Reference

```
#include <tstats.h>
```

Inheritance diagram for TStats:



### Public Member Functions

- [TStats](#) ()
- [TStats](#) (const [TStats](#) &orig)
- virtual [~TStats](#) ()
- void [add](#) (const [TStats](#) &otherStat)
- [TStats](#) \* [operator+](#) (const [TStats](#) &otherStat)
- [TStats](#) & [operator+=](#) (const [TStats](#) &otherStat)

### 7.21.1 Constructor & Destructor Documentation

7.21.1.1 `TStats::TStats ( )`

7.21.1.2 `TStats::TStats ( const TStats & orig )`

7.21.1.3 `TStats::~~TStats ( )` `[virtual]`

### 7.21.2 Member Function Documentation

7.21.2.1 `void TStats::add ( const TStats & otherStat )`

7.21.2.2 `TStats * TStats::operator+ ( const TStats & otherStat )`

7.21.2.3 `TStats & TStats::operator+= ( const TStats & otherStat )`

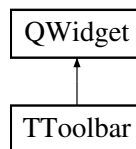
The documentation for this class was generated from the following files:

- [src/Data/tstats.h](#)
- [src/Data/tstats.cpp](#)

## 7.22 TToolbar Class Reference

```
#include <ttoolbar.h>
```

Inheritance diagram for TToolbar:



### Public Slots

- `void` [pauseTimer](#) ()
- `double` [incrementTimer](#) (int increment)

### Signals

- `void` [pauseClicked](#) ()

### Public Member Functions

- [TToolbar](#) (QWidget \*parent=0)
- [TToolbar](#) (const [TToolbar](#) &orig)
- `virtual` [~TToolbar](#) ()
- `void` [setProgression](#) (QString text)
- `void` [setLCDDisplayValue](#) (double value)



## 7.22.1 Constructor & Destructor Documentation

7.22.1.1 `TToolbar::TToolbar ( QWidget * parent = 0 )`

7.22.1.2 `TToolbar::TToolbar ( const TToolbar & orig )`

7.22.1.3 `virtual TToolbar::~~TToolbar ( ) [inline], [virtual]`

## 7.22.2 Member Function Documentation

7.22.2.1 `double TToolbar::incrementTimer ( int increment ) [slot]`

Add the value increment to the current value of the ldc display

Parameters

<i>increment</i>	the value of incrementation of the LDC display
------------------	--

7.22.2.2 `void TToolbar::pauseClicked ( ) [signal]`

Triggered whenever the user click on the pause/play button

Parameters

<i>inPause</i>	true if the state change to pause, false otherwise
----------------	--

7.22.2.3 `void TToolbar::pauseTimer ( ) [slot]`

Called whenever the user click on the pause/play button

7.22.2.4 `void TToolbar::setLCDDisplayValue ( double value ) [inline]`

7.22.2.5 `void TToolbar::setProgression ( QString text ) [inline]`

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

- [src/QTypingTest/Dialogs/Exercices/ttoolbar.h](#)
- [src/QTypingTest/Dialogs/Exercices/ttoolbar.cpp](#)

## 7.23 TUser Class Reference

```
#include <tuser.h>
```

### Public Member Functions

- [TUser](#) (QString pseudo= "")
- virtual [~TUser](#) ()
- QString [getPseudo](#) () const
- void [setPseudo](#) (QString pseudo)
- QHash< [date\\_exercice\\_\\*](#), [TResult](#) \* > [getPracticeHistory](#) () const

- [TProgression](#) \* [getProgression](#) ( ) const
- const [TStats](#) & [getStatistics](#) ( ) const
- void [oneMoreMistake](#) (const QChar &mistaken)
- QDateTime [addResult](#) ( [TExercise](#) \*exTyp, [TResult](#) \*exRes )

## 7.23.1 Constructor & Destructor Documentation

7.23.1.1 `TUser::TUser ( QString pseudo = " " )`

7.23.1.2 `TUser::~~TUser ( )` [virtual]

## 7.23.2 Member Function Documentation

7.23.2.1 `QDateTime TUser::addResult ( TExercise * exTyp, TResult * exRes )`

Add result to the history of user practice. Save the date, and the type of the exercise as a key, then add the result as value

Parameters

<i>exTyp</i>	the exercise done by the user
<i>exRes</i>	the result of the exercise

Returns

the time when this result was saved

7.23.2.2 `QHash<date_exercise_*, TResult*> TUser::getPracticeHistory ( ) const` [inline]

7.23.2.3 `TProgression* TUser::getProgression ( ) const` [inline]

7.23.2.4 `QString TUser::getPseudo ( ) const` [inline]

7.23.2.5 `const TStats& TUser::getStatistics ( ) const` [inline]

7.23.2.6 `void TUser::oneMoreMistake ( const QChar & mistaken )`

7.23.2.7 `void TUser::setPseudo ( QString pseudo )` [inline]

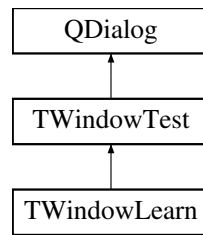
The documentation for this class was generated from the following files:

- [src/Data/tuser.h](#)
- [src/Data/tuser.cpp](#)

## 7.24 TWindowLearn Class Reference

```
#include <twindowlearn.h>
```

Inheritance diagram for TWindowLearn:



## Public Member Functions

- [TWindowLearn](#) (QString model, QWidget \*parent=0)
- [TWindowLearn](#) (QString model, int numberOfPages, QWidget \*parent=0)
- virtual [~TWindowLearn](#) ()

## Protected Member Functions

- void [updateClock](#) ()

## Additional Inherited Members

### 7.24.1 Constructor & Destructor Documentation

7.24.1.1 `TWindowLearn::TWindowLearn ( QString model, QWidget * parent = 0 )`

7.24.1.2 `TWindowLearn::TWindowLearn ( QString model, int numberOfPages, QWidget * parent = 0 )`

7.24.1.3 `virtual TWindowLearn::~~TWindowLearn ( )` [inline],[virtual]

### 7.24.2 Member Function Documentation

7.24.2.1 `void TWindowLearn::updateClock ( )` [inline],[protected],[virtual]

Called to update the clock of the toolbar

Implements [TWindowTest](#).

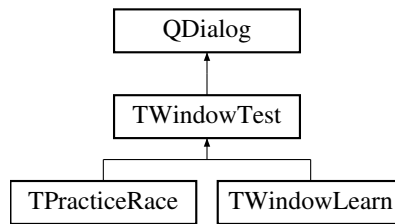
The documentation for this class was generated from the following files:

- src/QTypingTest/Dialogs/Exercices/[twindowlearn.h](#)
- src/QTypingTest/Dialogs/Exercices/[twindowlearn.cpp](#)

## 7.25 TWindowTest Class Reference

```
#include <twindowtest.h>
```

Inheritance diagram for TWindowTest:



## Public Slots

- void [saveResult](#) (TResult \*previousScore)
- void [beginExercise](#) ()
- void [exerciceFinished](#) (bool forced=false)
- void [pauseContinueExercise](#) ()
- virtual void [updateClock](#) ()=0
- void [updateToolbarProgression](#) ()

## Signals

- void [endOfExercise](#) (TResult \*exeRes, QTime timeRealised)
- void [closed](#) ()

## Public Member Functions

- [TWindowTest](#) (QWidget \*parent=0)
- [TWindowTest](#) (QString text, QWidget \*parent=0)
- [TWindowTest](#) (QString text, int numberOfPages, QWidget \*parent=0)
- [TToolbar toolbar](#) () const
- [TStackPages stackPages](#) () const
- virtual [~TWindowTest](#) ()

## Protected Member Functions

- void [keyPressEvent](#) (QKeyEvent \*ev)
- void [moveEvent](#) (QMoveEvent \*ev)
- void [closeEvent](#) (QCloseEvent \*ev)
- void [setupWidget](#) ()
- void [setupTimer](#) ()
- QString [getPageProgression](#) ()
- TResult \* [exerciceResult](#) ()
- void [setupShortcuts](#) ()
- void [setTimers](#) (int timeLimit)

## Protected Attributes

- [TToolbar topToolbar\\_](#)
- [TStackPages pages\\_](#)
- QList< TResult \* > [results\\_](#)

## 7.25.1 Constructor & Destructor Documentation

7.25.1.1 TWindowTest::TWindowTest ( QWidget \* *parent* = 0 ) [inline]

7.25.1.2 TWindowTest::TWindowTest ( QString *text*, QWidget \* *parent* = 0 ) [inline]

7.25.1.3 TWindowTest::TWindowTest ( QString *text*, int *numberOfPages*, QWidget \* *parent* = 0 ) [inline]

7.25.1.4 TWindowTest::~~TWindowTest ( ) [virtual]

## 7.25.2 Member Function Documentation

7.25.2.1 void TWindowTest::beginExercise ( ) [slot]

7.25.2.2 void TWindowTest::closed ( ) [signal]

Called whenever the exercise is forced to finish and no result/score is expected from the user

7.25.2.3 void TWindowTest::closeEvent ( QCloseEvent \* *ev* ) [protected]

Create a signal when the dialog close to warn the parent and reset the exercise

Parameters

<i>the</i>	close event
------------	-------------

7.25.2.4 void TWindowTest::endOfExercise ( TResult \* *exeRes*, QTime *timeRealised* ) [signal]

Triggerred when the exercise is 'officially' finished Either thanks to a debug shortcut or the text to copy is finished

Parameters

<i>exeRes</i>	the result of the exercise
<i>timeRealised</i>	the time realised to complete the exercise

7.25.2.5 void TWindowTest::exerciseFinished ( bool *forced* = false ) [slot]

Called whenever wa have the exerice is finished

Parameters

<i>a</i>	boolean to know if the user really ended the exercise or if he exited
----------	---

7.25.2.6 TResult \* TWindowTest::exerciseResult ( ) [protected]

Because each result is stored in a list, at the end of the rush the function needs to return a single result. This function sum up all the results and return the total sum.

Returns

the sum of all the result of each lines

#### 7.25.2.7 QString TWindowTest::getPageProgression ( ) [inline], [protected]

Return the progression as a fraction For example, if the user is at the first page and the exercise is composed of two pages, it will return "1/2"

##### Returns

the page progression

#### 7.25.2.8 void TWindowTest::keyPressEvent ( QKeyEvent \* *ev* ) [protected]

Instead of having to focus on each of the lines this function handle the keypress event for the whole dialog

##### Parameters

<i>ev</i>	the keyevent event
-----------	--------------------

#### 7.25.2.9 void TWindowTest::moveEvent ( QMoveEvent \* *ev* ) [protected]

Dialog keeps moving when changin the text of the label ...

##### Parameters

<i>the</i>	move event
------------	------------

Please dialog, stop moving when the text is changing !

#### 7.25.2.10 void TWindowTest::pauseContinueExercise ( ) [slot]

Called whenever the user click on the pause/continue button depending on the state, the button will react differently if the exercise is paused, it will resume it if the exercise is not paused, it will pause it

#### 7.25.2.11 void TWindowTest::saveResult ( TResult \* *previousScore* ) [slot]

To call whenever a line is finished and the next one must begin !

##### Parameters

<i>previousScore</i>	the score of the line that was finished
----------------------	---

#### 7.25.2.12 void TWindowTest::setTimers ( int *timeLimit* ) [protected]

Set the timers of the exercise and the values to update/calculate the time realised by the user

#### 7.25.2.13 void TWindowTest::setupShortcuts ( ) [protected]

Initiate all the shortcuts of the dialogs There are two type of shortcuts

- The 'hack' ones, that must be remove for the production

- The normal ones, to help the user being more efficient

7.25.2.14 `void TWindowTest::setupTimer ( )` [protected]

Setup the timer to update the chronometer

7.25.2.15 `void TWindowTest::setupWidget ( )` [protected]

Init function called as soon as the dialog is created setup the options of the dialog and create the lines depending on the given model

7.25.2.16 `TStackPages TWindowTest::stackPages ( ) const` [inline]

7.25.2.17 `TToolBar TWindowTest::toolBar ( ) const` [inline]

7.25.2.18 `virtual void TWindowTest::updateClock ( )` [pure virtual],[slot]

Called to update the clock of the toolbar

Implemented in [TWindowLearn](#).

7.25.2.19 `void TWindowTest::updateToolBarProgression ( )` [inline],[slot]

### 7.25.3 Member Data Documentation

7.25.3.1 `TStackPages TWindowTest::pages_` [protected]

7.25.3.2 `QList<TResult*> TWindowTest::results_` [protected]

7.25.3.3 `TToolBar TWindowTest::topToolBar_` [protected]

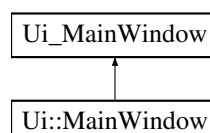
The documentation for this class was generated from the following files:

- `src/QTypingTest/Dialogs/twindowtest.h`
- `src/QTypingTest/Dialogs/twindowtest.cpp`

## 7.26 Ui\_MainWindow Class Reference

```
#include <ui_homepage.h>
```

Inheritance diagram for Ui\_MainWindow:



### Public Member Functions

- `void setupUi (QMainWindow *MainWindow)`
- `void retranslateUi (QMainWindow *MainWindow)`

## Public Attributes

- QAction \* [actionChange\\_user](#)
- QAction \* [actionOnline\\_competitoin](#)
- QAction \* [actionOthers](#)
- QAction \* [actionText\\_practice](#)
- QAction \* [actionMap\\_keyboard](#)
- QAction \* [actionLanguage](#)
- QAction \* [actionPreferences](#)
- QAction \* [actionChange\\_user\\_2](#)
- QAction \* [actionHomepage](#)
- QWidget \* [centralwidget](#)
- QWidget \* [verticalLayoutWidget](#)
- QVBoxLayout \* [verticalLayout](#)
- QHBoxLayout \* [horizontalLayout\\_2](#)
- QPushButton \* [button\\_home](#)
- QPushButton \* [button\\_learn](#)
- QPushButton \* [button\\_practice](#)
- QPushButton \* [button\\_stats](#)
- QPushButton \* [button\\_games](#)
- QGridLayout \* [gridLayout\\_2](#)
- QStackedWidget \* [stack\\_main](#)
- QWidget \* [page\\_learn](#)
- QWidget \* [page\\_home](#)
- QWidget \* [page\\_practice](#)
- QWidget \* [page\\_stats](#)
- QLabel \* [label\\_3](#)
- QWidget \* [page\\_games](#)
- QLabel \* [label\\_4](#)
- QMenuBar \* [menubar](#)
- QMenu \* [menuUser](#)
- QMenu \* [menuOptions](#)
- QMenu \* [menuAbout](#)
- QStatusBar \* [statusbar](#)

### 7.26.1 Member Function Documentation

7.26.1.1 void [Ui\\_MainWindow::retranslateUi](#) ( QMainWindow \* *MainWindow* ) [inline]

7.26.1.2 void [Ui\\_MainWindow::setupUi](#) ( QMainWindow \* *MainWindow* ) [inline]

### 7.26.2 Member Data Documentation

7.26.2.1 QAction\* [Ui\\_MainWindow::actionChange\\_user](#)

7.26.2.2 QAction\* [Ui\\_MainWindow::actionChange\\_user\\_2](#)

7.26.2.3 QAction\* [Ui\\_MainWindow::actionHomepage](#)

7.26.2.4 QAction\* [Ui\\_MainWindow::actionLanguage](#)

7.26.2.5 QAction\* [Ui\\_MainWindow::actionMap\\_keyboard](#)

7.26.2.6 QAction\* [Ui\\_MainWindow::actionOnline\\_competitoin](#)



- 7.26.2.7 QAction\* Ui\_MainWindow::actionOthers
- 7.26.2.8 QAction\* Ui\_MainWindow::actionPreferences
- 7.26.2.9 QAction\* Ui\_MainWindow::actionText\_practice
- 7.26.2.10 QPushButton\* Ui\_MainWindow::button\_games
- 7.26.2.11 QPushButton\* Ui\_MainWindow::button\_home
- 7.26.2.12 QPushButton\* Ui\_MainWindow::button\_learn
- 7.26.2.13 QPushButton\* Ui\_MainWindow::button\_practice
- 7.26.2.14 QPushButton\* Ui\_MainWindow::button\_stats
- 7.26.2.15 QWidget\* Ui\_MainWindow::centralwidget
- 7.26.2.16 QGridLayout\* Ui\_MainWindow::gridLayout\_2
- 7.26.2.17 QHBoxLayout\* Ui\_MainWindow::horizontalLayout\_2
- 7.26.2.18 QLabel\* Ui\_MainWindow::label\_3
- 7.26.2.19 QLabel\* Ui\_MainWindow::label\_4
- 7.26.2.20 QMenu\* Ui\_MainWindow::menuAbout
- 7.26.2.21 QMenuBar\* Ui\_MainWindow::menubar
- 7.26.2.22 QMenu\* Ui\_MainWindow::menuOptions
- 7.26.2.23 QMenu\* Ui\_MainWindow::menuUser
- 7.26.2.24 QWidget\* Ui\_MainWindow::page\_games
- 7.26.2.25 QWidget\* Ui\_MainWindow::page\_home
- 7.26.2.26 QWidget\* Ui\_MainWindow::page\_learn
- 7.26.2.27 QWidget\* Ui\_MainWindow::page\_practice
- 7.26.2.28 QWidget\* Ui\_MainWindow::page\_stats
- 7.26.2.29 QStackedWidget\* Ui\_MainWindow::stack\_main
- 7.26.2.30 QStatusBar\* Ui\_MainWindow::statusbar
- 7.26.2.31 QVBoxLayout\* Ui\_MainWindow::verticalLayout
- 7.26.2.32 QWidget\* Ui\_MainWindow::verticalLayoutWidget

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

- [src/ui/ui\\_homepage.h](#)



## Chapter 8

# File Documentation

### 8.1 doxygen.txt File Reference

### 8.2 etc/words/en.txt File Reference

### 8.3 etc/words/fr.txt File Reference

### 8.4 README.md File Reference

### 8.5 src/Data/texercice.cpp File Reference

```
#include "texercice.h"
```

### 8.6 src/Data/texercice.h File Reference

```
#include <QStringList>
#include <QDebug>
#include "tresult.h"
#include "Util/factory.h"
```

#### Classes

- class [TExercice](#)

### 8.7 src/Data/tkeys.cpp File Reference

```
#include "tkeys.h"
```

## 8.8 src/Data/tkeys.h File Reference

```
#include <qnamespace.h>
#include <QList>
```

### Classes

- class [TKeys](#)

## 8.9 src/Data/tpractical.cpp File Reference

```
#include "tpractical.h"
```

## 8.10 src/Data/tpractical.h File Reference

### Classes

- class [TPractical](#)

## 8.11 src/Data/tpractice.cpp File Reference

```
#include "tpractice.h"
```

## 8.12 src/Data/tpractice.h File Reference

```
#include <QStringList>
#include <QDebug>
```

### Classes

- class [TPractice](#)

## 8.13 src/Data/tprogression.cpp File Reference

```
#include "tprogression.h"
```

### Functions

- QDataStream & [operator<<](#) (QDataStream &out, const [TProgression](#) &prog)
- QDataStream & [operator>>](#) (QDataStream &in, [TProgression](#) &prog)
- bool [operator==](#) (const [TProgression](#) &prog1, const [TProgression](#) &prog2)

### 8.13.1 Function Documentation

8.13.1.1 `QDataStream& operator<< ( QDataStream & out, const TProgression & prog )`

8.13.1.2 `bool operator== ( const TProgression & prog1, const TProgression & prog2 )`

8.13.1.3 `QDataStream& operator>> ( QDataStream & in, TProgression & prog )`

## 8.14 src/Data/tprogression.h File Reference

```
#include <QDataStream>
```

### Classes

- class [TProgression](#)

### Functions

- `QDataStream & operator<< (QDataStream &out, const TProgression &prog)`
- `QDataStream & operator>> (QDataStream &in, TProgression &prog)`
- `bool operator== (const TProgression &prog1, const TProgression &prog2)`

### 8.14.1 Function Documentation

8.14.1.1 `QDataStream& operator<< ( QDataStream & out, const TProgression & prog )`

8.14.1.2 `bool operator== ( const TProgression & prog1, const TProgression & prog2 )`

8.14.1.3 `QDataStream& operator>> ( QDataStream & in, TProgression & prog )`

## 8.15 src/Data/tresult.cpp File Reference

```
#include "tresult.h"
```

## 8.16 src/Data/tresult.h File Reference

```
#include <QString>
#include "Util/htmlhelper.h"
```

### Classes

- class [TResult](#)

## 8.17 src/Data/tstats.cpp File Reference

```
#include "tstats.h"
```

## 8.18 src/Data/tstats.h File Reference

```
#include <QHash>
```

### Classes

- class [TStats](#)

## 8.19 src/Data/tuser.cpp File Reference

```
#include "tuser.h"
```

### Functions

- QDataStream & [operator<<](#) (QDataStream &out, const [TUser](#) &user)
- QDataStream & [operator>>](#) (QDataStream &in, [TUser](#) &user)

### 8.19.1 Function Documentation

8.19.1.1 QDataStream& [operator<<](#) ( QDataStream & *out*, const [TUser](#) & *user* )

8.19.1.2 QDataStream& [operator>>](#) ( QDataStream & *in*, [TUser](#) & *user* )

## 8.20 src/Data/tuser.h File Reference

```
#include <QString>
#include <QDateTime>
#include <QDebug>
#include <QHash>
#include "tprogression.h"
#include "texercice.h"
#include "tpractice.h"
#include "tresult.h"
#include "tstats.h"
```

### Classes

- struct [date\\_exercice\\_](#)
- class [TUser](#)

### Functions

- QDataStream & [operator<<](#) (QDataStream &out, const [TUser](#) &user)
- QDataStream & [operator>>](#) (QDataStream &in, [TUser](#) &user)
- bool [operator==](#) (const [TUser](#) &user1, const [TUser](#) &user2)

### 8.20.1 Function Documentation

8.20.1.1 `QDataStream& operator<< ( QDataStream & out, const TUser & user )`

8.20.1.2 `bool operator==( const TUser & user1, const TUser & user2 )` `[inline]`

8.20.1.3 `QDataStream& operator>> ( QDataStream & in, TUser & user )`

## 8.21 src/main.cpp File Reference

```
#include <QApplication>
#include <QDebug>
#include <QDir>
#include <time.h>
#include <QFile>
#include "QTypingTest/thomepage.h"
```

### Functions

- `int main (int argc, char *argv[])`

### 8.21.1 Function Documentation

8.21.1.1 `int main ( int argc, char * argv[] )`

## 8.22 src/QTypingTest/Dialogs/Exercices/tlabel.cpp File Reference

```
#include "tlabel.h"
```

## 8.23 src/QTypingTest/Dialogs/Exercices/tlabel.h File Reference

```
#include <QWidget>
#include <QLabel>
#include <QDebug>
#include <QFont>
#include "Util/htmlhelper.h"
```

### Classes

- `class TLabel`

## 8.24 src/QTypingTest/Dialogs/Exercices/tpage.cpp File Reference

```
#include "tpage.h"
```

## 8.25 src/QTypingTest/Dialogs/Exercices/tpage.h File Reference

```
#include <QWidget>
#include <QLineEdit>
#include <QVBoxLayout>
#include <QKeyEvent>
#include <QApplication>
#include <QValidator>
#include <QList>
#include "Data/tresult.h"
#include "Data/tkeys.h"
#include "Util/factory.h"
#include "tlabel.h"
```

### Classes

- class [TPage](#)

## 8.26 src/QTypingTest/Dialogs/Exercices/tpacticerace.cpp File Reference

```
#include "tpacticerace.h"
```

## 8.27 src/QTypingTest/Dialogs/Exercices/tpacticerace.h File Reference

```
#include <QWidget>
#include <QDebug>
#include "QTypingTest/Dialogs/twindowtest.h"
#include "Data/texercice.h"
#include "Data/tresult.h"
```

### Classes

- class [TPracticeRace](#)

## 8.28 src/QTypingTest/Dialogs/Exercices/tstackpages.cpp File Reference

```
#include <qt5/QtCore/qlogging.h>
#include <qt5/QtCore/qstringlist.h>
#include "tstackpages.h"
```



## 8.29 src/QTypingTest/Dialogs/Exercices/tstackpages.h File Reference

```
#include <QWidget>
#include <QStackedWidget>
#include <QStringList>
#include <QList>
#include <QKeyEvent>
#include <QDebug>
#include "Data/tresult.h"
#include "QTypingTest/Dialogs/Exercices/tpage.h"
```

### Classes

- class [TStackPages](#)

## 8.30 src/QTypingTest/Dialogs/Exercices/ttoolbar.cpp File Reference

```
#include <qt5/QtCore/qnamespace.h>
#include <qt5/QtCore/qlogging.h>
#include "ttoolbar.h"
```

## 8.31 src/QTypingTest/Dialogs/Exercices/ttoolbar.h File Reference

```
#include <QWidget>
#include <QCheckBox>
#include <QLCDNumber>
#include <QLabel>
#include <QTimer>
#include <QSpacerItem>
#include <QHBoxLayout>
#include <qnamespace.h>
#include <QDebug>
```

### Classes

- class [TToolbar](#)

## 8.32 src/QTypingTest/Dialogs/Exercices/twindowlearn.cpp File Reference

```
#include "twindowlearn.h"
```

## 8.33 src/QTypingTest/Dialogs/Exercices/twindowlearn.h File Reference

```
#include <QWidget>
#include "QTypingTest/Dialogs/twindowtest.h"
```

## Classes

- class [TWindowLearn](#)

### 8.34 src/QTypingTest/Dialogs/twindowtest.cpp File Reference

```
#include <qt5/QtCore/qlogging.h>
#include "twindowtest.h"
```

### 8.35 src/QTypingTest/Dialogs/twindowtest.h File Reference

```
#include <QDialog>
#include <QWidget>
#include <QTime>
#include <QShortcut>
#include <QKeySequence>
#include <QVBoxLayout>
#include <QKeyEvent>
#include <QDebug>
#include "Exercices/tstackpages.h"
#include "Exercices/ttoolbar.h"
#include "Data/tresult.h"
#include "Data/texercice.h"
#include "Util/factory.h"
```

## Classes

- class [TWindowTest](#)

### 8.36 src/QTypingTest/Learn/learnpage.cpp File Reference

```
#include "learnpage.h"
```

## 8.37 src/QTypingTest/Learn/learnpage.h File Reference

```
#include <QWidget>
#include <QPushButton>
#include <QMessageBox>
#include <QGridLayout>
#include <QDebug>
#include <QVector>
#include <QScrollArea>
#include "Data/tpractice.h"
#include "Data/texercice.h"
#include "Data/tresult.h"
#include "Util/filehelper.h"
#include "Util/htmlhelper.h"
#include "QTypingTest/Dialogs/Exercices/twindowlearn.h"
#include "QTypingTest/thomepage.h"
```

### Classes

- class [LearnPage](#)

### Macros

- #define [MAX\\_COL](#) 6

#### 8.37.1 Macro Definition Documentation

8.37.1.1 #define [MAX\\_COL](#) 6

## 8.38 src/QTypingTest/Practice/practicepage.cpp File Reference

```
#include "practicepage.h"
```

## 8.39 src/QTypingTest/Practice/practicepage.h File Reference

```
#include <QWidget>
#include <QPushButton>
#include <QGridLayout>
#include <QMessageBox>
#include "QTypingTest/Dialogs/Exercices/tpracticerace.h"
```

### Classes

- class [PracticePage](#)

## 8.40 src/QTypingTest/thomepage.cpp File Reference

```
#include "thomepage.h"
```

## 8.41 src/QTypingTest/thomepage.h File Reference

```
#include <QWidget>
#include <QMainWindow>
#include <QVector>
#include <QPushButton>
#include <QDebug>
#include <QHBoxLayout>
#include "Learn/learnpage.h"
#include "Practice/practicepage.h"
#include "Data/tpractice.h"
#include "ui/ui_homepage.h"
#include "Data/tuser.h"
```

### Classes

- class [THomePage](#)

## 8.42 src/ui/ui\_homepage.h File Reference

```
#include <QtCore/QVariant>
#include <QtWidgets/QAction>
#include <QtWidgets/QApplication>
#include <QtWidgets/QButtonGroup>
#include <QtWidgets/QGridLayout>
#include <QtWidgets/QHBoxLayout>
#include <QtWidgets/QHeaderView>
#include <QtWidgets/QLabel>
#include <QtWidgets/QMainWindow>
#include <QtWidgets/QMenu>
#include <QtWidgets/QMenuBar>
#include <QtWidgets/QPushButton>
#include <QtWidgets/QStackedWidget>
#include <QtWidgets/QStatusBar>
#include <QtWidgets/QVBoxLayout>
#include <QtWidgets/QWidget>
```

### Classes

- class [Ui\\_MainWindow](#)
- class [Ui::MainWindow](#)

### Namespaces

- [Ui](#)

## 8.43 src/Util/factory.cpp File Reference

```
#include <qt5/QtCore/qstringlist.h>
#include <qt5/QtCore/qlogging.h>
#include "factory.h"
```

### Functions

- int [randomNextSpace](#) ()
- bool [wordsContains](#) (QStringList words, QString contains)
- QStringList [readFile](#) (QString fileName)
- bool [isValidWord](#) (QString word, QString availableLetters)

### Variables

- const int [numberOfWords](#) = 40

#### 8.43.1 Function Documentation

8.43.1.1 bool [isValidWord](#) ( QString *word*, QString *availableLetters* )

8.43.1.2 int [randomNextSpace](#) ( )

A little helper to generate random position of space in text

##### Returns

A random number between 2 and 9 (the next position of the space)

8.43.1.3 QStringList [readFile](#) ( QString *fileName* )

8.43.1.4 bool [wordsContains](#) ( QStringList *words*, QString *contains* )

Little helper to check if at least each chars of the contains QString is contained in one of the words. For example, with the words ("fly","my") and the string "ty", the function will return false since "t" is contained in none of the words "fly" and "my". If we had ("fly","my","try") the function would return true since t is in "try" and "y" is in at least of the three words

##### Parameters

<i>words</i>	the found words
<i>contains</i>	the chars that must be in the words

##### Returns

if ALL the chars in the string contains are in the words

#### 8.43.2 Variable Documentation

8.43.2.1 const int [numberOfWords](#) = 40

## 8.44 src/Util/factory.h File Reference

```
#include <QStringList>
#include <QDebug>
#include <QFile>
#include <QTextStream>
#include <algorithm>
#include "filehelper.h"
```

### Namespaces

- [factory](#)

### Functions

- QString [factory::generateLearning](#) (QStringList mainLetter, QStringList [allLetters](#))
- QString [factory::generatePractice](#) (QStringList letters, bool onlyRealWords=true)
- QString [factory::generateFromLetters](#) (QStringList letterList, int length=50)
- QString [factory::generateWords](#) (QString authorizedLetters, QString language, QString mainLetters="")
- QStringList [factory::findExistingWords](#) (QString authorizedLetters, QString filename, QString mustContain="")
- QString [factory::selectRandomElement](#) (QStringList listString)
- int [factory::findClosestSpace](#) (const QString &search, int indexStart)
- QStringList [factory::splitText](#) (QString toSplit, int numberOfSplit)

## 8.45 src/Util/filehelper.cpp File Reference

```
#include "filehelper.h"
```

### Variables

- const QString [WORDS\\_PATH](#) = "words"
- const QString [WORDS\\_EXT](#) = ".txt"
- const QString [ETC\\_PATH](#) = "etc"

### 8.45.1 Variable Documentation

8.45.1.1 const QString [ETC\\_PATH](#) = "etc"

8.45.1.2 const QString [WORDS\\_EXT](#) = ".txt"

8.45.1.3 const QString [WORDS\\_PATH](#) = "words"

## 8.46 src/Util/filehelper.h File Reference

```
#include <QString>
```

## Namespaces

- [file](#)

## Functions

- QString [file::getWordsPath](#) (QString lang)
- QString [file::getPadlockIcon](#) (bool open)
- QString [file::getStylesheet](#) (QString stylesheetname)

## Variables

- const QString [ETC\\_PATH](#)
- const QString [WORDS\\_PATH](#)
- const QString [WORDS\\_EXT](#)

### 8.46.1 Variable Documentation

8.46.1.1 const QString [ETC\\_PATH](#)

8.46.1.2 const QString [WORDS\\_EXT](#)

8.46.1.3 const QString [WORDS\\_PATH](#)

## 8.47 src/Util/htmlhelper.cpp File Reference

```
#include "htmlhelper.h"
```

## Functions

- QString [surroundOfTags](#) (QStringList tags, QString attributes)

### 8.47.1 Function Documentation

8.47.1.1 QString [surroundOfTags](#) ( QStringList *tags*, QString *attributes* )

Little helper to create tags into tags and return a regex-ready string to add the text needed in the center

#### Parameters

<i>tags</i>	the list of tag to add around a char
-------------	--------------------------------------

**Returns**

a regex-ready string to add the needed text in the center of the tags

## 8.48 src/Util/htmlhelper.h File Reference

```
#include <QStringList>
#include <QRegExp>
#include <QDebug>
```

**Namespaces**

- [html](#)

**Functions**

- [QString html::addTagToChars](#) (QString strToTag, QString tag, QString attributes="")
- [QString html::removeTag](#) (QString &strToClean, QString tag="([>]\*)")
- [QString html::addTags](#) (QString &strToTag, QString tags, QString attributes="")
- [int html::getAbsoluteCharPosition](#) (QString string, int relativePos)

## 8.49 tests/TestLearnDialog/testlearndialog.cpp File Reference

```
#include "QTypingTest/Dialogs/Exercices/twindowlearn.h"
#include <QtTest/QtTest>
#include <QApplication>
#include <time.h>
#include "testlearndialog.moc"
```

**Classes**

- class [TestLearnDialog](#)

## 8.50 tests/TestPracticeRace/testpracticrace.cpp File Reference

```
#include "QTypingTest/Dialogs/Exercices/tpracticrace.h"
#include "Data/tresult.h"
#include <QtTest/QtTest>
#include <QApplication>
#include <QTime>
#include <qt5/QtWidgets/MessageBox>
#include <time.h>
#include "testpracticrace.moc"
```

**Classes**

- class [TestPracticeRace](#)



## 8.51 tests/TestProgression/testprogression.cpp File Reference

```
#include <QFile>
#include <QDataStream>
#include "../src/Data/tprogression.h"
#include <QtTest/QtTest>
#include "testprogression.moc"
```

### Classes

- class [TestProgression](#)

## 8.52 tests/TestUser/testuser.cpp File Reference

```
#include <QtTest/QtTest>
#include <QFile>
#include <QDebug>
#include <QVector>
#include <QDataStream>
#include "../src/Data/texercice.h"
#include "../src/Data/tresult.h"
#include "../src/Data/tuser.h"
#include "testuser.moc"
```

### Classes

- class [TestTUser](#)

## 8.53 tests/TestUtils/testutils.cpp File Reference

```
#include <QtTest/QtTest>
#include <QDebug>
#include <QStringList>
#include <QDir>
#include <time.h>
#include "../src/Util/htmlhelper.h"
#include "../src/Util/factory.h"
#include "../src/Util/filehelper.h"
#include "testutils.moc"
```

### Classes

- class [TestUtils](#)

### Variables

- const QStringList [allLetters](#) = QStringList("abcdefghijklmnopqrstuvwxyz")

### 8.53.1 Variable Documentation

8.53.1.1 `const QStringList allLetters = QStringList("abcdefghijklmnopqrstuvwxyz")`

# Index

- ~LearnPage
  - LearnPage, [18](#)
- ~PracticePage
  - PracticePage, [19](#)
- ~TExercice
  - TExercice, [21](#)
- ~THomePage
  - THomePage, [22](#)
- ~TKeys
  - TKeys, [23](#)
- ~TLabel
  - TLabel, [24](#)
- ~TPractical
  - TPractical, [27](#)
- ~TPractice
  - TPractice, [28](#)
- ~TPracticeRace
  - TPracticeRace, [29](#)
- ~TProgression
  - TProgression, [29](#)
- ~TResult
  - TResult, [30](#)
- ~TStackPages
  - TStackPages, [32](#)
- ~TStats
  - TStats, [34](#)
- ~TToolbar
  - TToolbar, [35](#)
- ~TUser
  - TUser, [36](#)
- ~TWindowLearn
  - TWindowLearn, [37](#)
- ~TWindowTest
  - TWindowTest, [39](#)
- AZERTY
  - TPractice, [27](#)
- actionChange\_user
  - Ui\_MainWindow, [42](#)
- actionChange\_user\_2
  - Ui\_MainWindow, [42](#)
- actionHomepage
  - Ui\_MainWindow, [42](#)
- actionLanguage
  - Ui\_MainWindow, [42](#)
- actionMap\_keyboard
  - Ui\_MainWindow, [42](#)
- actionOnline\_competitoin
  - Ui\_MainWindow, [42](#)
- actionOthers
  - Ui\_MainWindow, [42](#)
- actionPreferences
  - Ui\_MainWindow, [43](#)
- actionText\_practice
  - Ui\_MainWindow, [43](#)
- add
  - TStats, [34](#)
- addPage
  - TStackPages, [32](#)
- addResult
  - TUser, [36](#)
- addTagToChars
  - html, [15](#)
- addTags
  - html, [15](#)
- allLetters
  - testutils.cpp, [60](#)
- avdvanceExIndex
  - TProgression, [30](#)
- BASIC
  - TPractical, [26](#)
- beginExercice
  - TStackPages, [32](#)
  - TWindowTest, [39](#)
- buildExercice
  - TExercice, [22](#)
- button\_games
  - Ui\_MainWindow, [43](#)
- button\_home
  - Ui\_MainWindow, [43](#)
- button\_learn
  - Ui\_MainWindow, [43](#)
- button\_practice
  - Ui\_MainWindow, [43](#)
- button\_stats
  - Ui\_MainWindow, [43](#)
- CHUCK\_NORRIS
  - TPractical, [26](#)
- centralwidget
  - Ui\_MainWindow, [43](#)
- closeEvent
  - TWindowTest, [39](#)
- closed
  - TWindowTest, [39](#)
- createPage
  - TPracticeRace, [29](#)
- currentPage
  - TStackPages, [32](#)

- currentUser\_
  - THomePage, [22](#)
- DIFFICULTY
  - TPractical, [26](#)
- date\_exercice\_, [17](#)
  - dateResult, [17](#)
  - exercice, [17](#)
- dateResult
  - date\_exercice\_, [17](#)
- doxygen.txt, [45](#)
- EASY
  - TPractical, [26](#)
- ETC\_PATH
  - filehelper.cpp, [56](#)
  - filehelper.h, [57](#)
- EXERCICE\_TYPE
  - TExercice, [21](#)
- endExercice
  - LearnPage, [18](#)
- endOfExercice
  - TWindowTest, [39](#)
- endedPage
  - TPage, [25](#)
- eraseAnswer
  - TPage, [25](#)
- eraseOverflow
  - TLabel, [24](#)
- etc/words/en.txt, [45](#)
- etc/words/fr.txt, [45](#)
- exercice
  - date\_exercice\_, [17](#)
- exerciceFinished
  - TWindowTest, [39](#)
- exerciceGenerator
  - TPracticeRace, [29](#)
- exerciceResult
  - TWindowTest, [39](#)
- exerciceStarted
  - TStackPages, [32](#)
- factory, [11](#)
  - findClosestSpace, [11](#)
  - findExistingWords, [11](#)
  - generateFromLetters, [12](#)
  - generateLearning, [12](#)
  - generatePractice, [12](#)
  - generateWords, [13](#)
  - selectRandomElement, [13](#)
  - splitText, [13](#)
- factory.cpp
  - isValidWord, [55](#)
  - numberOfWords, [55](#)
  - randomNextSpace, [55](#)
  - readFile, [55](#)
  - wordsContains, [55](#)
- file, [14](#)
  - getPadlockIcon, [14](#)
  - getStylesheet, [14](#)
  - getWordsPath, [14](#)
- filehelper.cpp
  - ETC\_PATH, [56](#)
  - WORDS\_EXT, [56](#)
  - WORDS\_PATH, [56](#)
- filehelper.h
  - ETC\_PATH, [57](#)
  - WORDS\_EXT, [57](#)
  - WORDS\_PATH, [57](#)
- findClosestSpace
  - factory, [11](#)
- findExistingWords
  - factory, [11](#)
- generateFromLetters
  - factory, [12](#)
- generateLearning
  - factory, [12](#)
- generatePractice
  - factory, [12](#)
- generateWords
  - factory, [13](#)
- getAbsoluteCharPosition
  - html, [15](#)
- getAllLettersTo
  - TPractice, [28](#)
- getCorrectKeysStrokes
  - TResult, [31](#)
- getCorrectWords
  - TResult, [31](#)
- getInstance
  - TPractice, [28](#)
- getLastExerciceIndex
  - TProgression, [30](#)
- getLetterList
  - TPractice, [28](#)
- getLettersAt
  - TPractice, [28](#)
- getPadlockIcon
  - file, [14](#)
- getPageProgression
  - TWindowTest, [39](#)
- getPracticeHistory
  - TUser, [36](#)
- getProgression
  - TUser, [36](#)
- getPseudo
  - TUser, [36](#)
- getResult
  - TPage, [25](#)
- getResume
  - TResult, [31](#)
- getStatistics
  - TUser, [36](#)
- getStylesheet
  - file, [14](#)
- getText
  - TStackPages, [32](#)

- getTotalKeysStokres
  - TResult, 31
- getTotalWords
  - TResult, 31
- getUser
  - THomePage, 22
- getWPM
  - TResult, 31
- getWordsPath
  - file, 14
- getWrongKeysStrokes
  - TResult, 31
- getWrongWords
  - TResult, 31
- gridLayout\_2
  - Ui\_MainWindow, 43
- HARD
  - TPractical, 26
- horizontalLayout\_2
  - Ui\_MainWindow, 43
- html, 14
  - addTagToChars, 15
  - addTags, 15
  - getAbsoluteCharPosition, 15
  - removeTag, 16
- htmlhelper.cpp
  - surroundOfTags, 57
- incrCorrectKeystrokes
  - TResult, 31
- incrWrongKeyStrokes
  - TResult, 31
- incrementTimer
  - TToolbar, 35
- isValidLetter
  - TKeys, 23
- isValidWord
  - factory.cpp, 55
- KEYBOARD\_LAYOUT
  - TPpractice, 27
- keyPressEvent
  - TWindowTest, 40
- keyPressed
  - TStackPages, 32
- LEARNING
  - TExercice, 21
- label\_3
  - Ui\_MainWindow, 43
- label\_4
  - Ui\_MainWindow, 43
- lauchExercice
  - LearnPage, 18
- LearnPage, 17
  - ~LearnPage, 18
  - endExercice, 18
  - lauchExercice, 18
  - LearnPage, 18
  - resetExercice, 18
- learnpage.h
  - MAX\_COL, 53
- MAX\_COL
  - learnpage.h, 53
- MEDIUM
  - TPractical, 26
- main
  - main.cpp, 49
- main.cpp
  - main, 49
- menuAbout
  - Ui\_MainWindow, 43
- menuOptions
  - Ui\_MainWindow, 43
- menuUser
  - Ui\_MainWindow, 43
- menubar
  - Ui\_MainWindow, 43
- moveEvent
  - TWindowTest, 40
- NORMAL
  - TPractical, 26
- nextChar
  - TLabel, 24
- nextPage
  - TStackPages, 33
- numberOfPages
  - TStackPages, 33
- numberOfWords
  - factory.cpp, 55
- oneMoreMistake
  - TUser, 36
- operator<<
  - tprogression.cpp, 47
  - tprogression.h, 47
  - tuser.cpp, 48
  - tuser.h, 49
- operator>>
  - tprogression.cpp, 47
  - tprogression.h, 47
  - tuser.cpp, 48
  - tuser.h, 49
- operator+
  - TResult, 31
  - TStats, 34
- operator+=
  - TResult, 31
  - TStats, 34
- operator==
  - tprogression.cpp, 47
  - tprogression.h, 47
  - tuser.h, 49
- PRACTICING

- TExercice, 21
- page\_games
  - Ui\_MainWindow, 43
- page\_home
  - Ui\_MainWindow, 43
- page\_learn
  - Ui\_MainWindow, 43
- page\_practice
  - Ui\_MainWindow, 43
- page\_stats
  - Ui\_MainWindow, 43
- pageEnded
  - TStackPages, 33
- pages\_
  - TWindowTest, 41
- pauseClicked
  - TToolbar, 35
- pauseContinueExercise
  - TWindowTest, 40
- pauseTimer
  - TToolbar, 35
- PracticePage, 18
  - ~PracticePage, 19
  - PracticePage, 19
  - saveExerciseResult, 19
  - startAgainstTime, 19
- previousChar
  - TLabel, 24
- QWERTY
  - TPractice, 27
- QWERTZ
  - TPractice, 27
- README.md, 45
- randomNextSpace
  - factory.cpp, 55
- readFile
  - factory.cpp, 55
- removeTag
  - html, 16
- resetExercise
  - LearnPage, 18
- results\_
  - TWindowTest, 41
- retranslateUi
  - Ui\_MainWindow, 42
- saveExerciseResult
  - PracticePage, 19
- saveResult
  - TWindowTest, 40
- selectRandomElement
  - factory, 13
- setCorrectKeysStrokes
  - TResult, 31
- setFirst
  - TLabel, 24
- setFollowingPage
  - TStackPages, 33
- setLCDDisplayValue
  - TToolbar, 35
- setLastExerciceIndex
  - TProgression, 30
- setProgression
  - TToolbar, 35
- setPseudo
  - TUser, 36
- setTimers
  - TWindowTest, 40
- setWrongKeysStrokes
  - TResult, 31
- setupShortcuts
  - TWindowTest, 40
- setupTimer
  - TWindowTest, 41
- setupUi
  - Ui\_MainWindow, 42
- setupWidget
  - TWindowTest, 41
- splitText
  - factory, 13
- src/Data/texercice.cpp, 45
- src/Data/texercice.h, 45
- src/Data/tkeys.cpp, 45
- src/Data/tkeys.h, 46
- src/Data/tpactical.cpp, 46
- src/Data/tpactical.h, 46
- src/Data/tpactice.cpp, 46
- src/Data/tpactice.h, 46
- src/Data/tprogression.cpp, 46
- src/Data/tprogression.h, 47
- src/Data/tresult.cpp, 47
- src/Data/tresult.h, 47
- src/Data/tstats.cpp, 47
- src/Data/tstats.h, 48
- src/Data/tuser.cpp, 48
- src/Data/tuser.h, 48
- src/QTypingTest/Dialogs/Exercices/tlabel.cpp, 49
- src/QTypingTest/Dialogs/Exercices/tlabel.h, 49
- src/QTypingTest/Dialogs/Exercices/tpage.cpp, 49
- src/QTypingTest/Dialogs/Exercices/tpage.h, 50
- src/QTypingTest/Dialogs/Exercices/tpacticerace.cpp, 50
- src/QTypingTest/Dialogs/Exercices/tpacticerace.h, 50
- src/QTypingTest/Dialogs/Exercices/tstackpages.cpp, 50
- src/QTypingTest/Dialogs/Exercices/tstackpages.h, 51
- src/QTypingTest/Dialogs/Exercices/ttoolbar.cpp, 51
- src/QTypingTest/Dialogs/Exercices/ttoolbar.h, 51
- src/QTypingTest/Dialogs/Exercices/twindowlearn.cpp, 51
- src/QTypingTest/Dialogs/Exercices/twindowlearn.h, 51
- src/QTypingTest/Dialogs/twindowtest.cpp, 52
- src/QTypingTest/Dialogs/twindowtest.h, 52
- src/QTypingTest/Learn/learnpage.cpp, 52
- src/QTypingTest/Learn/learnpage.h, 53
- src/QTypingTest/Practice/practicepage.cpp, 53

- src/QTypingTest/Practice/practicepage.h, 53
- src/QTypingTest/thomepage.cpp, 54
- src/QTypingTest/thomepage.h, 54
- src/Util/factory.cpp, 55
- src/Util/factory.h, 56
- src/Util/filehelper.cpp, 56
- src/Util/filehelper.h, 56
- src/Util/htmlhelper.cpp, 57
- src/Util/htmlhelper.h, 58
- src/main.cpp, 49
- src/ui/ui\_homepage.h, 54
- stack\_main
  - Ui\_MainWindow, 43
- stackPages
  - TWindowTest, 41
- startAgainstTime
  - PracticePage, 19
- startedPage
  - TPage, 25
- statusbar
  - Ui\_MainWindow, 43
- surroundOfTags
  - htmlhelper.cpp, 57
- TEXT\_PRACTICING
  - TExercice, 21
- TExercice, 21
  - ~TExercice, 21
  - buildExercice, 22
  - EXERCICE\_TYPE, 21
  - LEARNING, 21
  - PRACTICING, 21
  - TEXT\_PRACTICING, 21
  - TExercice, 21
- THomePage, 22
  - ~THomePage, 22
  - currentUser\_, 22
  - getUser, 22
  - THomePage, 22
- TKeys, 23
  - ~TKeys, 23
  - isValidLetter, 23
  - TKeys, 23
  - validLetters, 23
- TLabel, 23
  - ~TLabel, 24
  - eraseOverflow, 24
  - nextChar, 24
  - previousChar, 24
  - setFirst, 24
  - TLabel, 24
- TPage, 24
  - endedPage, 25
  - eraseAnswer, 25
  - getResult, 25
  - startedPage, 25
  - TPage, 25
  - text, 25
  - typingAnswer, 26
  - update, 26
  - updateAsFirst, 26
- TPractical, 26
  - ~TPractical, 27
  - BASIC, 26
  - CHUCK\_NORRIS, 26
  - DIFFICULTY, 26
  - EASY, 26
  - HARD, 26
  - MEDIUM, 26
  - NORMAL, 26
  - TPractical, 27
- TPractice, 27
  - ~TPractice, 28
  - AZERTY, 27
  - getAllLettersTo, 28
  - getInstance, 28
  - getLetterList, 28
  - getLettersAt, 28
  - KEYBOARD\_LAYOUT, 27
  - QWERTY, 27
  - QWERTZ, 27
- TPracticeRace, 28
  - ~TPracticeRace, 29
  - createPage, 29
  - exerciceGenerator, 29
  - TPracticeRace, 28, 29
  - updateClock, 29
- TProgression, 29
  - ~TProgression, 29
  - avdvanceExIndex, 30
  - getLastExerciceIndex, 30
  - setLastExerciceIndex, 30
  - TProgression, 29
- TResult, 30
  - ~TResult, 30
  - getCorrectKeysStrokes, 31
  - getCorrectWords, 31
  - getResume, 31
  - getTotalKeysStokres, 31
  - getTotalWords, 31
  - getWPM, 31
  - getWrongKeysStrokes, 31
  - getWrongWords, 31
  - incrCorrectKeystrokes, 31
  - incrWrongKeyStrokes, 31
  - operator+, 31
  - operator+=", 31
  - setCorrectKeysStrokes, 31
  - setWrongKeysStrokes, 31
  - TResult, 30
  - updateWPM, 31
- TStackPages, 31
  - ~TStackPages, 32
  - addPage, 32
  - beginExercice, 32
  - currentPage, 32
  - exerciceStarted, 32

- getText, 32
  - keyPressed, 32
  - nextPage, 33
  - numberOfPages, 33
  - pageEnded, 33
  - setFollowingPage, 33
  - TStackPages, 32
  - textFinished, 33
- TStats, 33
  - ~TStats, 34
  - add, 34
  - operator+, 34
  - operator+=, 34
  - TStats, 34
- TToolbar, 34
  - ~TToolbar, 35
  - incrementTimer, 35
  - pauseClicked, 35
  - pauseTimer, 35
  - setLCDDisplayValue, 35
  - setProgression, 35
  - TToolbar, 35
- TUser, 35
  - ~TUser, 36
  - addResult, 36
  - getPracticeHistory, 36
  - getProgression, 36
  - getPseudo, 36
  - getStatistics, 36
  - oneMoreMistake, 36
  - setPseudo, 36
  - TUser, 36
- TWindowLearn, 36
  - ~TWindowLearn, 37
  - TWindowLearn, 37
  - updateClock, 37
- TWindowTest, 37
  - ~TWindowTest, 39
  - beginExercise, 39
  - closeEvent, 39
  - closed, 39
  - endOfExercise, 39
  - exerciseFinished, 39
  - exerciseResult, 39
  - getPageProgression, 39
  - keyPressEvent, 40
  - moveEvent, 40
  - pages\_, 41
  - pauseContinueExercise, 40
  - results\_, 41
  - saveResult, 40
  - setTimers, 40
  - setupShortcuts, 40
  - setupTimer, 41
  - setupWidget, 41
  - stackPages, 41
  - TWindowTest, 39
  - toolBar, 41
  - topToolBar\_, 41
  - updateClock, 41
  - updateToolBarProgression, 41
- TestLearnDialog, 19
- TestPracticeRace, 20
- TestProgression, 20
- TestTUser, 20
- TestUtils, 20
- tests/TestLearnDialog/testlearndialog.cpp, 58
- tests/TestPracticeRace/testpracticerace.cpp, 58
- tests/TestProgression/testprogression.cpp, 59
- tests/TestUser/testuser.cpp, 59
- tests/TestUtils/testutils.cpp, 59
- testutils.cpp
  - allLetters, 60
- text
  - TPage, 25
- textFinished
  - TStackPages, 33
- toolBar
  - TWindowTest, 41
- topToolBar\_
  - TWindowTest, 41
- tprogression.cpp
  - operator<<, 47
  - operator>>, 47
  - operator==, 47
- tprogression.h
  - operator<<, 47
  - operator>>, 47
  - operator==, 47
- tuser.cpp
  - operator<<, 48
  - operator>>, 48
- tuser.h
  - operator<<, 49
  - operator>>, 49
  - operator==, 49
- typingAnswer
  - TPage, 26
- Ui, 16
- Ui::MainWindow, 18
- Ui\_MainWindow, 41
  - actionChange\_user, 42
  - actionChange\_user\_2, 42
  - actionHomepage, 42
  - actionLanguage, 42
  - actionMap\_keyboard, 42
  - actionOnline\_competitoir, 42
  - actionOthers, 42
  - actionPreferences, 43
  - actionText\_practice, 43
  - button\_games, 43
  - button\_home, 43
  - button\_learn, 43
  - button\_practice, 43
  - button\_stats, 43
  - centralwidget, 43



- gridLayout\_2, [43](#)
- horizontalLayout\_2, [43](#)
- label\_3, [43](#)
- label\_4, [43](#)
- menuAbout, [43](#)
- menuOptions, [43](#)
- menuUser, [43](#)
- menubar, [43](#)
- page\_games, [43](#)
- page\_home, [43](#)
- page\_learn, [43](#)
- page\_practice, [43](#)
- page\_stats, [43](#)
- retranslateUi, [42](#)
- setupUi, [42](#)
- stack\_main, [43](#)
- statusbar, [43](#)
- verticalLayout, [43](#)
- verticalLayoutWidget, [43](#)
- update
  - TPage, [26](#)
- updateAsFirst
  - TPage, [26](#)
- updateClock
  - TPracticeRace, [29](#)
  - TWindowLearn, [37](#)
  - TWindowTest, [41](#)
- updateToolBarProgression
  - TWindowTest, [41](#)
- updateWPM
  - TResult, [31](#)
- validLetters
  - TKeys, [23](#)
- verticalLayout
  - Ui\_MainWindow, [43](#)
- verticalLayoutWidget
  - Ui\_MainWindow, [43](#)
- WORDS\_EXT
  - filehelper.cpp, [56](#)
  - filehelper.h, [57](#)
- WORDS\_PATH
  - filehelper.cpp, [56](#)
  - filehelper.h, [57](#)
- wordsContains
  - factory.cpp, [55](#)