



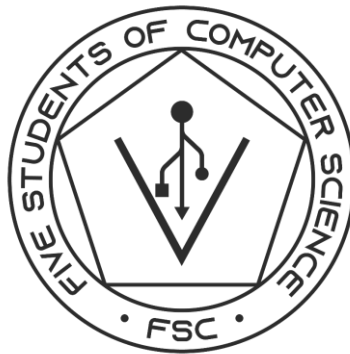
Informatica e Comunicazione Digitale - (TA)

A.A. 2018/19

STRUMENTALMENTE - DOCUMENTAZIONE

DOCUMENTO DI REALIZZAZIONE





FSC — Five Students of Computer Science

Copyright © 2019, FSC

Concesso in licenza secondo i termini della Licenza Apache, versione 2.0 (la "Licenza"); è proibito usare questo file se non in conformità alla Licenza. Una copia della Licenza è disponibile all'indirizzo:

<http://www.apache.org/licenses/LICENSE-2.0>

Se non richiesto dalla legislazione vigente o concordato per iscritto, il software distribuito nei termini della Licenza è distribuito *"così com'è"*, senza garanzie o condizioni di alcun tipo, esplicite o implicite. Consultare la Licenza per il testo specifico che regola le autorizzazioni e le limitazioni previste dalla medesima.

Sommario

1	Le tecnologie utilizzate	4
1.1	I linguaggi	4
1.2	Il codice	4
2	Avviare dal codice sorgente	6
2.1	Il Makefile.....	6
3	Documentazione del codice	8
3.1	Classes.....	8
3.2	Funzioni	8
3.2.1	Main	11
3.2.2	openMobileNavigation()	11
3.2.3	setLinks(firstSlideNumber, links)	11
3.2.4	updateBlindAudio(newSource)	12
3.2.5	initialize(initial, base, totalNumberOfSlides).....	12
3.2.6	changeTopic(topicName, [base])	13
3.2.7	initializeQuiz().....	13
3.2.8	changeQuizSlide(finalSlide).....	13
3.2.9	playStopAudio(audioTagId, buttonRef, stopButtonId).....	14
3.2.10	openModal(content, [options], [windowIcon])	14
3.2.11	showQuizDialog(nomeQuiz, score, total, return_link)	14
3.2.12	showExitFromQuizDialog(toOpen)	14
3.2.13	showExitDialog().....	15
3.2.14	warnIfIncomplete(previousQuizId, previousQuizName, topicToOpenName, callback)	15
3.2.15	setUpTitleBar().....	15
3.2.16	openInBrowser(link).....	16
3.2.17	getUsername().....	16
3.2.18	setUsername(newUsername).....	16
3.2.19	getQuiz(id).....	16
3.2.20	generateRandomQuestions()	17
3.2.21	quizLoad(id)	17

3.2.22	quizVerify(fail_return_link, success_return_link)	17
3.2.23	quizCompare().....	17
3.2.24	quizSelectNextPage(failedPage, passedPage)	18
3.2.25	script_load()	18
3.2.26	verify_and_store().....	18
3.2.27	correct_chord()	18
3.2.28	script_load()	18
3.2.29	verify_and_store().....	19
3.2.30	correct_chord()	19
3.2.31	script_load()	19
3.2.32	verify_and_store().....	19
3.2.33	correct_chord()	19
3.2.34	selectFirstColumn(j).....	20
3.2.35	controlFirstColumn(j)	20

1 Le tecnologie utilizzate

1.1 I linguaggi

Il sistema StrumentalMente è scritto utilizzando l'**HTML5** (il che porta all'utilizzo del **CSS3** per definire le varie regole di stile) e il **JavaScript** (con la presenza di pochi file scritti in *TypeScript*, un dialetto del JavaScript e l'uso di alcune funzioni *jQuery*). L'intera applicazione è basata su *Node.js*, principalmente per il pacchetto **Electron**, che consente di utilizzare i precedenti linguaggi, in combinazione con il motore di rendering *Chromium*, per creare delle applicazioni *desktop*.

1.2 Il codice

Il codice sorgente di StrumentalMente è fondamentalmente suddiviso in tre file principali (escludendo i file HTML e CSS utilizzati per creare l'aspetto grafico dell'applicazione). Tali file sono:

app.ts

È il file principale dell'applicazione. Contiene il processo principale ("*main process*") di Electron e gestisce, quindi, tutti gli eventi principali di StrumentalMente, tra cui:

- L'apertura dell'applicazione e il *rendering* della prima finestra
- La chiusura dell'applicazione e le relative peculiarità di alcuni sistemi operativi (si pensi alla possibilità di ricreare la finestra appena chiusa su MacOS)
- L'apertura di finestre di dialogo
- L'apertura di finestra secondarie

Esegue i suoi compiti utilizzando la classe `Main` definita nel file `StrumentalMente.ts`.

StrumentalMente.ts

Contiene la definizione della classe principale che gestisce gli eventi del *main process* di Electron.

render.js

È il file principale del *rendering* dell'applicazione. Contiene tutte le funzioni da eseguire nel "*rendering process*" (processo di *rendering*) di Electron.

main.js

Contiene tutte le funzioni che servono a dare dinamicità alle pagine HTML dell'applicazione.

quiz.js**chord-module.js**

Questo modulo contiene le definizioni basilari utilizzati dai moduli inerenti gli accordi.

accordi_basso.js

Contiene il comportamento dei quiz sugli accordi di basso.

accordi_chitarra.js

Contiene il comportamento dei quiz sugli accordi di chitarra.

accordi_piano.js

Contiene il comportamento dei quiz sugli accordi di pianoforte.

modal-dialog-module.js

Questo modulo contiene la definizione del comportamento delle finestre di dialogo.

vendor/jquery.maphilight.js

Un modulo non sviluppato dal team. Permette di evidenziare le aree di una mappa interattiva (tag <map> dell'HTML). Il file è scritto in JQuery ed è stato lievemente modificato dal membro del team Andrea Esposito.

2 Avviare dal codice sorgente

Affinchè si possa avviare StrumentalMente dal codice sorgente è necessario che siano installati `Node.js` e `yarn`. Successivamente, è necessario installare le dipendenze del progetto, eseguendo il seguente comando nella *directory* in cui è situato il codice sorgente (cartella `src/` della *repository*):

```
yarn install
```

Dopo aver installato tutte le dipendenze, è possibile avviare l'applicazione utilizzando il comando seguente all'interno della *directory* del codice sorgente:

```
yarn start
```

2.1 Il Makefile

Per comodità, si è creato un `Makefile` per eseguire facilmente tutte le operazioni sul codice. Il comando `make` è da eseguire all'interno della *directory* principale della *repository*.

Il file `Makefile` contiene le seguenti regole:

(all)

Avvia StrumentalMente.

yarn-install

Installa yarn.

install

Installa le dipendenze del progetto (con l'opzione `-force` che forza l'eventuale reinstallazione).

start

Avvia StrumentalMente.

deploy

Compila StrumentalMente in un eseguibile per la piattaforma su cui il comando è stato eseguito.

docs

Ricompile tutta la documentazione partendo dai sorgenti LaTeX.

jsdoc

Ricompile la documentazione del codice in un file LaTeX.

bib2html

Converte la bibliografia in formato BibTeX in HTML e la importa automaticamente nell'apposita sezione della pagina "about".

docx-to-html

Converte i file DOCX in file HTML.

3 Documentazione del codice

La presente documentazione è stata generata in modo semi-automatico da *JSDoc*, *JSDoc to Markdown* e *Pandoc*.

3.1 Classes

Main

3.2 Funzioni

openMobileNavigation()

Apri la navbar in modalità "mobile". Questa funzione è mantenuta solo per consentire un eventuale eccessivo ridimensionamento della finestra.

setLinks(firstSlideNumber, links)

Cambia i link e i nomi dell'argomento precedente e quello successivo a quello attuale

updateBlindAudio(newSource)

Aggiorna la sorgente dell'audio per ciechi in base alla sezione corrente.

Il comportamento che segue è il seguente (descritto in pseudocodice):

```
if è_in_pausa AND cambia_slide then
  pausa()
elseif è_in_play AND cambia_slide then
  play()
elseif è_finito AND non_stoppato AND cambia_slide then
  play()
elseif cambia_argomento then
  pausa()
```

initialize(initial, base, totalNumberOfSlides)

Funzione che, al caricamento della pagina, si occupa di impostare il numero di tag section presenti all'interno della pagina nella memoria locale del browser, di impostare come sezione visibile corrente la prima (sempre all'interno della memoria locale) e di nascondere tutti i tag section successivi al primo.

changeTopic(topicName, [base])

Cambia l'argomento correntemente mostrato.

initializeQuiz()

Inizializza la pagina del quiz.

changeQuizSlide(finalSlide)

Cambia la slide del quiz attualmente mostrata.

playStopAudio(audioTagId, buttonRef, stopButtonId)

Permette di avviare, mettere in pausa o stoppare un audio.

openModal(content, [options], [windowIcon])

Apri una finestra modale mostrante il contenuto richiesto.

showQuizDialog(nomeQuiz, score, total, return_link)

Mostra il dialogo con il punteggio dei quiz.

showExitFromQuizDialog(toOpen)

Mostra il dialogo di richiesta di conferma di uscita dal quiz.

showExitDialog()

Mostra il dialogo di richiesta di conferma di uscita.

warnIfIncomplete(previousQuizId, previousQuizName, topicToOpenName, callback)

Mostra un messaggio all'utente se il quiz propedeutico all'argomento scelto non è stato completato. Se l'utente conferma di voler proseguire, viene effettuata l'azione richiesta, altrimenti non si attua alcuna azione.

setUpTitleBar()

Gestisce gli eventi della titlebar.

Questa funzione gestisce gli eventi (riduci a icona, massimizza/minimizza, chiudi) che sono accessibili tramite la titlebar.

openInBrowser(link)

Apri un link nel browser predefinito.

getUsername()

Ritorna l'username collegato a StrumentalMente.

setUsername(newUsername)

Imposta l'username dell'utente.

getQuiz(id)

Ottiene il risultato del quiz scelto

generateRandomQuestions()

La funzione genera una permutazione casuale dei numeri naturali nell'intervallo [0,3]. Ogni numero è utilizzato per indicizzare il vettore delle risposte (di ogni domanda), quindi la permutazione corrisponde alla permutazione delle risposte a ogni domanda.

quizLoad(id)

Funzione di caricamento dei quiz che inizializza il punteggio dell'utente, il punteggio massimo del quiz e richiama la funzione initializeQuiz. In caso di quiz che riguardano gli accordi, richiama la funzione di caricamento degli stessi.

quizVerify(return_link)

Richiama la funzione di Check delle risposte inoltre, se si tratta di un quiz che riguarda gli accordi, richiama anche la funzione `verify_and_store()`. Infine, visualizza la finestra di dialogo che mostra il punteggio ottenuto dall'utente su quello massimo del quiz e richiede se uscire dal quiz o verificare le risposte date.

quizCompare()

Mostra gli eventuali errori commessi dall'utente mostrando le risposte corrette e le eventuali risposte sbagliate. Nel caso di un quiz sugli accordi, mostra l'accordo corretto al fianco di quello errato.

script_load()

Seleziona un numero casuale compreso tra 1 e 24 e ne sceglie il relativo accordo dall'array degli accordi, lo imposta come domanda nel quiz e lo inserisce in un altro array che contiene gli accordi "pescati", ripetendo il procedimento per le 5 domande totali richieste nel quiz.

verify_and_store()

Verifica che le selezioni effettuate dall'utente siano corrette in base all'accordo presentatogli e memorizza in un array se la selezione è corretta (1) o non corretta (0), disabilita le checkbox analizzate e aumenta il punteggio (in caso di accordo corretto).

correct_chord()

In base al numero di accordo che l'utente doveva riprodurre, ripristina la sequenza di selezioni corretta negli schemi e ne blocca le modifiche.

script_load()

Seleziona un numero casuale compreso tra 1 e 24 e ne sceglie il relativo accordo dall'array degli accordi, lo imposta come domanda nel quiz e lo inserisce in un altro array che contiene gli accordi "pescati", ripetendo il procedimento per le 5 domande totali richieste nel quiz.

verify_and_store()

Verifica che le selezioni effettuate dall'utente siano corrette in base all'accordo presentatogli e memorizza in un array se la selezione è corretta (1) o non corretta (0), disabilita le checkbox analizzate e aumenta il punteggio (in caso di accordo corretto).

correct_chord()

In base al numero di accordo che l'utente doveva riprodurre, ripristina la sequenza di selezioni corretta negli schemi e ne blocca le modifiche.

script_load()

Seleziona un numero casuale compreso tra 1 e 24 e ne sceglie il relativo accordo dall'array degli accordi, lo imposta come domanda nel quiz e lo inserisce in un altro array che contiene gli accordi "pescati", ripetendo il procedimento per le 5 domande totali richieste nel quiz.

verify_and_store()

Verifica che le selezioni effettuate dall'utente siano corrette in base all'accordo presentatogli e memorizza in un array se la selezione è corretta (1) o non corretta (0), disabilita le checkbox analizzate e aumenta il punteggio (in caso di accordo corretto).

correct_chord()

In base al numero di accordo che l'utente doveva riprodurre, ripristina la sequenza di selezioni corretta negli schemi e ne blocca le modifiche.

selectFirstColumn(j)

Seleziona (o deseleziona) tutte le checkbox della prima colonna nell'accordo j-esimo (passato come parametro) in base al valore della prima checkbox dell'accordo j-esimo, ovvero: se il valore della prima checkbox è true (selezionato) deseleziona tutta la colonna, se è false la seleziona.

controlFirstColumn(j)

Modifica il nome del bottone che permette all'utente di applicare o rimuovere il barrè nei quiz in modo da renderlo coerente con lo stato delle checkbox.

3.2.1 Main

Kind: global class

3.2.1.1 new Main()

La classe contenente la logica principale dell'applicazione.

3.2.2 openMobileNavigation()

Apri la navbar in modalità "mobile". Questa funzione è mantenuta solo per consentire un eventuale eccessivo ridimensionamento della finestra.

Kind: global function

3.2.3 setLinks(firstSlideNumber, links)

Cambia i link e i nomi dell'argomento precedente e quello successivo a quello attuale

Kind: global function

Param	Type	Description
-------	------	-------------

firstSlideNumber	number	Il numero della prima slide dell'argomento corrente
links	Object	Le nuove impostazioni e link
links.previous	String	Il nome della pagina precedente
links.previousLink	String	Il link della pagina precedente (il nome del file <i>senza</i> l'estensione)
links.next	String	Il nome della pagina successiva
links.nextLink	String	Il link della pagina successiva (il nome del file <i>senza</i> l'estensione)

3.2.4 updateBlindAudio(newSource)

Aggiorna la sorgente dell'audio per ciechi in base alla sezione corrente. Il comportamento che segue è il seguente (descritto in pseudocodice): if è_in_pausa AND cambia_slide then pausa() elseif è_in_play AND cambia_slide then play() elseif è_finito AND non_stoppato AND cambia_slide then play() elseif cambia_argomento then pausa()

Kind: global function

Param	Type	Description
newSource	String	La nuova sorgente per l'audio

3.2.5 initialize(initial, base, totalNumberOfSlides)

Funzione che, al caricamento della pagina, si occupa di impostare il numero di tag section presenti all'interno della pagina nella memoria locale del browser, di impostare come sezione visibile corrente la prima (sempre all'interno della memoria locale) e di nascondere tutti i tag sections successivi al primo.

Kind: global function

Param	Type	Default	Description
initial	String		Il primo argomento
base	String	./	La cartella in cui sono situati i file degli argomenti (default: ./)
totalNumberOfSlides	number		Il numero totale di pagine per la sezione.

3.2.5.1 initialize~changeSlide(slide)

La funzione, in base al valore assunto da slide cambia la sezione corrente in quella richiesta. Inoltre, si occupa di aggiornare il numero della slide corrente nella memoria temporanea del browser. Inoltre, in base al numero di slide, si occupa di rendere visibili (o nascondere) i relativi pulsanti di spostamento (avanti con id next, indietro con id back e quiz con id quiz).

Kind: inner method of initialize

Param	Type	Description
slide	number	Il numero della slide da aprire.

3.2.6 changeTopic(topicName, [base])

Cambia l'argomento correntemente mostrato.

Kind: global function

Param	Type	Default	Description
topicName	String		Il prossimo argomento
[base]	String	./	La cartella in cui è situato il file dell'argomento

3.2.7 initializeQuiz()

Inizializza la pagina del quiz.

Kind: global function

3.2.8 changeQuizSlide(finalSlide)

Cambia la slide del quiz attualmente mostrata.

Kind: global function

Param	Type	Description
finalSlide	number	La slide da aprire in seguito alla richiesta di variazione della slide. Tale valore deve essere compreso nell'intervallo $[0, n]$, dove n è il numero di slide presenti nella pagina.

3.2.9 playStopAudio(audioTagId, buttonRef, stopButtonId)

Permette di avviare, mettere in pausa o stoppare un audio.

Kind: global function

Param	Type	Description
audioTagId	String	L'ID dell'elemento <audio> da controllare
buttonRef	HTMLElement	Un riferimento al bottone che richiama questa funzione
stopButtonId	String	L'ID del bottone di Stop.

3.2.10 openModal(content, [options], [windowIcon])

Apri una finestra modale mostrante il contenuto richiesto.

Kind: global function

Param	Type	Default	Description
content	String		Il link (assoluto o relativo) da aprire
[options]	Object		Le opzioni della nuova finestra
[windowIcon]	String	./assets/icon.ico	L'icona della finestra modale

3.2.11 showQuizDialog(nomeQuiz, score, total, return_link)

Mostra il dialogo con il punteggio dei quiz.

Kind: global function

Param	Type	Description
nomeQuiz	String	Il nome del quiz.
score	number	Il punteggio ottenuto.
total	number	Il punteggio totale possibile.
return_link	String	Il file da aprire se è cliccato il tasto 'Ok'. Il percorso è relativo rispetto alla cartella principale.

3.2.12 showExitFromQuizDialog(toOpen)

Mostra il dialogo di richiesta di conferma di uscita dal quiz.

Kind: global function

Param	Type	Description
toOpen	String	Il file da aprire se è cliccato il tasto 'Sì'. Il percorso è relativo rispetto alla cartella principale.

3.2.13 showExitDialog()

Mostra il dialogo di richiesta di conferma di uscita.

Kind: global function

3.2.14 warnIfIncomplete(previousQuizId, previousQuizName, topicToOpenName, callback)

Mostra un messaggio all'utente se il quiz propedeutico all'argomento scelto non è stato completato. Se l'utente conferma di voler proseguire, viene effettuata l'azione richiesta, altrimenti non si attua alcuna azione.

Kind: global function

Param	Type	Description
previousQuizId	String	L'id del quiz propedeutico
previousQuizName	String	Il nome del quiz (da comunicare all'utente)
topicToOpenName	String	Il nome dell'argomento che si vuole aprire
callback	*	La funzione da eseguire se l'utente accetta di proseguire.

3.2.15 setUpTitleBar()

Gestisce gli eventi della titlebar. Questa funzione gestisce gli eventi (riduci a icona, massimizza/minimizza, chiudi) che sono accessibili tramite la titlebar.

Kind: global function

- setUpTitleBar()
 - ~init()
 - ~toggleMaxRestoreButtons()

3.2.15.1 setUpTitleBar~init()

Inizializza la titlebar.

Kind: inner method of setUpTitleBar

3.2.15.2 setUpTitleBar~toggleMaxRestoreButtons()

Cicla tra i bottoni di massimizzazione e diminimizzazione della finestra alternativamente

Kind: inner method of setUpTitleBar

3.2.16 openInBrowser(link)

Apri un link nel browser predefinito.

Kind: global function

Param	Type	Description
link	String	Il link da aprire

3.2.17 getUsername()

Ritorna l'username collegato a StrumentalMente.

Kind: global function

3.2.18 setUsername(newUsername)

Imposta l'username dell'utente.

Kind: global function

Param	Type	Description
newUsername	String	Il nuovo username.

3.2.19 getQuiz(id)

Ottiene il risultato del quiz scelto

Kind: global function

Param	Type	Description
-------	------	-------------

id **String** L'id del quiz di cui interessa il risultato.

3.2.20 **generateRandomQuestions()**

La funzione genera una permutazione casuale dei numeri naturali nell'intervallo [0,3]. Ogni numero è utilizzato per indicizzare il vettore delle risposte (di ogni domanda), quindi la permutazione corrisponde alla permutazione delle risposte a ogni domanda.

Kind: global function

3.2.21 **quizLoad(id)**

Funzione di caricamento dei quiz che inizializza il punteggio dell'utente, il punteggio massimo del quiz e richiama la funzione initializeQuiz. In caso di quiz che riguardano gli accordi, richiama la funzione di caricamento degli stessi.

Kind: global function

Param	Type	Description
id	String	Indica quale quiz si sta svolgendo per permettere al programma di memorizzarne l'esito.

3.2.22 **quizVerify(fail_return_link, success_return_link)**

Richiama la funzione di Check delle risposte inoltre, se si tratta di un quiz che riguarda gli accordi, richiama anche la funzione verify_and_store(). Infine, visualizza la finestra di dialogo che mostra il punteggio ottenuto dall'utente su quello massimo del quiz e richiede se uscire dal quiz o verificare le risposte date.

Kind: global function

Param	Type	Description
fail_return_link	String	Indica in quale pagina si deve ritornare al completamento dei quiz se non lo si è superato.
success_return_link	String	Indica in quale pagina si deve ritornare al completamento dei quiz se lo si è superato.

3.2.23 **quizCompare()**

Mostra gli eventuali errori commessi dall'utente mostrando le risposte corrette e le eventuali risposte sbagliate. Nel caso di un quiz sugli accordi, mostra l'accordo corretto al fianco di quello errato.

Kind: global function

3.2.24 **quizSelectNextPage(failedPage, passedPage)**

Apri una pagina basandosi sul risultato del quiz in cui la presente funzione viene richiamata.

Kind: global function

Param	Type	Description
failedPage	String	La pagina da aprire se il quiz non è stato superato
passedPage	String	La pagina da aprire se il quiz è stato superato

3.2.25 **script_load()**

Seleziona un numero casuale compreso tra 1 e 24 e ne sceglie il relativo accordo dall'array degli accordi, lo imposta come domanda nel quiz e lo inserisce in un altro array che contiene gli accordi "pescati", ripetendo il procedimento per le 5 domande totali richieste nel quiz.

Kind: global function

3.2.26 **verify_and_store()**

Verifica che le selezioni effettuate dall'utente siano corrette in base all'accordo presentatogli e memorizza in un array se la selezione è corretta (1) o non corretta (0), disabilita le checkbox analizzate e aumenta il punteggio (in caso di accordo corretto).

Kind: global function

3.2.27 **correct_chord()**

In base al numero di accordo che l'utente doveva riprodurre, ripristina la sequenza di selezioni corretta negli schemi e ne blocca le modifiche.

Kind: global function

3.2.28 **script_load()**

Seleziona un numero casuale compreso tra 1 e 24 e ne sceglie il relativo accordo dall'array degli accordi, lo imposta come domanda nel quiz e lo inserisce in un altro array che contiene gli accordi "pescati", ripetendo il procedimento per le 5 domande totali richieste nel quiz.

Kind: global function

3.2.29 verify_and_store()

Verifica che le selezioni effettuate dall'utente siano corrette in base all'accordo presentatogli e memorizza in un array se la selezione è corretta (1) o non corretta (0), disabilita le checkbox analizzate e aumenta il punteggio (in caso di accordo corretto).

Kind: global function

3.2.30 correct_chord()

In base al numero di accordo che l'utente doveva riprodurre, ripristina la sequenza di selezioni corretta negli schemi e ne blocca le modifiche.

Kind: global function

3.2.31 script_load()

Seleziona un numero casuale compreso tra 1 e 24 e ne sceglie il relativo accordo dall'array degli accordi, lo imposta come domanda nel quiz e lo inserisce in un altro array che contiene gli accordi "pescati", ripetendo il procedimento per le 5 domande totali richieste nel quiz.

Kind: global function

3.2.32 verify_and_store()

Verifica che le selezioni effettuate dall'utente siano corrette in base all'accordo presentatogli e memorizza in un array se la selezione è corretta (1) o non corretta (0), disabilita le checkbox analizzate e aumenta il punteggio (in caso di accordo corretto).

Kind: global function

3.2.33 correct_chord()

In base al numero di accordo che l'utente doveva riprodurre, ripristina la sequenza di selezioni corretta negli schemi e ne blocca le modifiche.

Kind: global function

3.2.34 **selectFirstColumn(j)**

Seleziona (o deseleziona) tutte le checkbox della prima colonna nell'accordo j-esimo (passato come parametro) in base al valore della prima checkbox dell'accordo j-esimo, ovvero: se il valore della prima checkbox è true (selezionato) deseleziona tutta la colonna, se è false la seleziona.

Kind: global function

Param	Type	Description
j	number	Indica il numero di slide su cui la funzione deve operare.

3.2.35 **controlFirstColumn(j)**

Modifica il nome del bottone che permette all'utente di applicare o rimuovere il barrè nei quiz in modo da renderlo coerente con lo stato delle checkbox.

Kind: global function

Param	Type	Description
j	number	Indica il numero di slide su cui la funzione deve operare.

