

Test Case Specification

AM-GP

Versione	Cambiamenti	Autore
0.2	Aggiunta casi di test per funzionalità da 1 a 3	Pietro Negri
0.4	Aggiunta casi di test per funzionalità da 4 a 5	Mafalda Ingenito
0.6	Aggiunta casi di test per funzionalità da 6 a 7	Antonio Trovato
1.0	Sistemazione Struttura del documento	Pietro Negri, Giovanni Renzulli
1.1	Aggiunta test case per test di unità	Mafalda Ingenito, Antonio Trovato

Indice generale

Test Cases per Funzionalità.....	5
1.1 Login.....	5
Scelta delle Categorie:.....	5
Parametro: ID.....	5
Parametro: Password.....	5
2.1 Invio Messaggio.....	9
Scelta delle Categorie:.....	9
Parametro: Destinatario.....	9
Parametro: Oggetto.....	9
Parametro: Corpo.....	9
3.1 Salva Setup.....	13
Scelta delle Categorie:.....	13
Parametro: Carico Aerodinamico Anteriore.....	13
Parametro: Carico Aerodinamico Posteriore.....	13
Parametro: Campanatura Anteriore.....	13
Parametro: Campanatura Posteriore.....	13
Parametro: Convergenza Anteriore.....	14
Parametro: Convergenza Posteriore.....	14
Parametro: Pressione dei Freni.....	14
Parametro: Barra Antirollio Posteriore.....	14
Parametro: Barra Antirollio Anteriore.....	15
Parametro dell'ambiente: Circuito.....	15
4.1 Modifica Setup.....	27
Scelta delle Categorie:.....	27
Parametro: ID.....	27
Parametro: Carico Aerodinamico Anteriore.....	27
Parametro: Carico Aerodinamico Posteriore.....	27
Parametro: Campanatura Anteriore.....	28
Parametro: Campanatura Posteriore.....	28
Parametro: Convergenza Anteriore.....	28
Parametro: Convergenza Posteriore.....	28
Parametro: Pressione dei Freni.....	29
Parametro: Barra Antirollio Posteriore.....	29
Parametro: Barra Antirollio Anteriore.....	29
5.1 Elimina Setup.....	41
Scelta delle Categorie:.....	41
Parametro: ID.....	41
6.1 Visualizzazione Circuito.....	43
Scelta delle Categorie:.....	43
Parametro: ID.....	43
7.1 Visualizzazione Setup.....	45
Scelta delle Categorie:.....	45
Parametro: ID.....	45
Test Cases per Test di Unità.....	48
1.1 Rettilineo.....	48
Scelta delle Categorie:.....	48
Parametro: Nome.....	48
Parametro: Lunghezza.....	48
2.1 Curva.....	53

Scelta delle Categorie:.....	53
Parametro: Nome.....	53
Parametro: Angolo.....	53
3.1 Circuito.....	58
Scelta delle Categorie:.....	58
Parametro: Nome.....	58
Parametro: Sede.....	58
Parametro: Lunghezza.....	58
Parametro: Meteo.....	58
Parametro: TPM.....	59
Parametro: RPM.....	59
Parametro: Umidità.....	59
Parametro: NumeroGiri.....	59
Parametro: Curve.....	60
Parametro: RETTILINEI.....	60
4.1 Pilota.....	78
Scelta delle Categorie:.....	78
Parametro: NumeroPole.....	78
Parametro: NumeroVittorie.....	78
Parametro: NumeroPiazzamenti.....	78
Parametro: NumeroRitiri.....	78
5.1 Setup.....	83
Scelta delle Categorie:.....	83
Parametro: CaricoAreodinamicoAnteriore.....	83
Parametro: CaricoAreodinamicoPosteriore.....	84
Parametro: CampanaturaAnteriore.....	84
Parametro: CampanaturaPosteriore.....	84
Parametro: PressioneFreni.....	84
Parametro: BarraAntirollioAnteriore.....	85
Parametro: BarraAntirollioPosteriore.....	85
Parametro: CampanaturaPosteriore.....	85
6.1 Tecnico.....	95
Scelta delle Categorie:.....	95
Parametro: SetupList.....	95
Parametro: Setup.....	96

Test Cases per Funzionalità

1.1 Login

Scelta delle Categorie:

Parametro: ID

Parametro	ID
Formato	
Categorie	Formato, Esistenza
Formato [ID_FRM]	1 Non rispetta il formato [error] 2 Rispetta il formato [ID_FRM_OK]
Esistente [EST]	1 Non esiste nel database [if ID_FRM_OK and PW_FRM_OK][error] 2 Esiste nel database [if ID_FRM_OK and PW_FRM_OK][EST_OK]

Parametro: Password

Parametro	Password
Formato	
Categorie	Formato, Match
Formato [PW_FRM]	1 Non rispetta il formato [error] 2 Rispetta il formato [PW_FRM_OK]
Match [MTCH]	1 Non matcha con la password associata [if EST_OK] [error] 2 Matcha con la password associata [if EST_OK] [MTCH_OK]

In base a questa scelta delle categorie, le possibili combinazioni sono:

ID_FRM	EST	PW_FRM	MTCH	Oracolo
1	//	//	//	error
2	//	1	//	error
2	1	2		error
2	2	2	1	error
2	2	2	2	MTCH_OK, ID_FRM_OK, PW_FRM_OK, EST_OK

I test case generati sono quindi:

Test Case ID: TC 1.1				
Precondizioni:				
L'utente si trova sulla pagina di login				
Flusso di Eventi:				
<ul style="list-style-type: none">L'utente inserisce i seguenti dati:<table><tr><td>ID</td><td></td></tr><tr><td>Password</td><td></td></tr></table>L'utente preme sul pulsante "Accedi"	ID		Password	
ID				
Password				
Oracolo:				
L'accesso non è avvenuto perché i campi obbligatori sono vuoti.				

Test Case ID: TC 1.2				
Precondizioni:				
L'utente si trova sulla pagina di login				
Flusso di Eventi:				
<ul style="list-style-type: none">L'utente inserisce i seguenti dati:<table><tr><td>ID</td><td>aas</td></tr><tr><td>Password</td><td></td></tr></table>L'utente preme sul pulsante "Accedi"	ID	aas	Password	
ID	aas			
Password				
Oracolo:				
L'accesso non è avvenuto perché il campo ID non rispetta il formato.				

Test Case ID: TC 1.3					
Precondizioni:					
L'utente si trova sulla pagina di login					
Flusso di Eventi:					
<ul style="list-style-type: none"> L'utente inserisce i seguenti dati: <table border="1"> <tr> <td>ID</td><td>AAAAAaABCDfERV</td></tr> <tr> <td>Password</td><td>fdsfs</td></tr> </table> L'utente preme sul pulsante "Accedi" 		ID	AAAAAaABCDfERV	Password	fdsfs
ID	AAAAAaABCDfERV				
Password	fdsfs				
Oracolo:					
L'accesso non è avvenuto perché la password non rispetta il formato.					

Test Case ID: TC 1.4					
Precondizioni:					
L'utente si trova sulla pagina di login					
Flusso di Eventi:					
<ul style="list-style-type: none"> L'utente inserisce i seguenti dati: <table border="1"> <tr> <td>ID</td><td>AAAAAaABCDfEVG</td></tr> <tr> <td>Password</td><td>SalveATuttiRagazzi32</td></tr> </table> L'utente preme sul pulsante "Accedi" 		ID	AAAAAaABCDfEVG	Password	SalveATuttiRagazzi32
ID	AAAAAaABCDfEVG				
Password	SalveATuttiRagazzi32				
Oracolo:					
L'accesso non è avvenuto perché l'ID non è presente nel DB					

Test Case ID: TC 1.5					
Precondizioni:					
L'utente si trova sulla pagina di login					
Flusso di Eventi:					
<ul style="list-style-type: none"> L'utente inserisce i seguenti dati: <table border="1"> <tr> <td>ID</td><td>AAAAAaABCDfERV</td></tr> <tr> <td>Password</td><td>SalveATuttiRagazzi31</td></tr> </table> L'utente preme sul pulsante "Accedi" 		ID	AAAAAaABCDfERV	Password	SalveATuttiRagazzi31
ID	AAAAAaABCDfERV				
Password	SalveATuttiRagazzi31				
Oracolo:					
L'accesso non è avvenuto perché la password non corrisponde quella associata all'ID					

Test Case ID: TC 1.6					
Precondizioni:					
L'utente si trova sulla pagina di login					
Flusso di Eventi:					
<ul style="list-style-type: none"> L'utente inserisce i seguenti dati: <table border="1"> <tr> <td>ID</td><td>AAAAAaABCDfERV</td></tr> <tr> <td>Password</td><td>SalveATuttiRagazzi32</td></tr> </table> L'utente preme sul pulsante "Accedi" 		ID	AAAAAaABCDfERV	Password	SalveATuttiRagazzi32
ID	AAAAAaABCDfERV				
Password	SalveATuttiRagazzi32				
Oracolo:					
L'accesso è avvenuto con successo.					

2.1 Invio Messaggio

Scelta delle Categorie:

Parametro: Destinatario

Per quanto riguarda il Destinatario, il Sistema propone già all'utente una serie di possibili destinatari presenti all'interno del sistema. Non potendo scegliere di inserire direttamente l'ID del destinatario, l'utente non può commettere degli errori relativi al suo formato o alla sua presenza nel DB.

Parametro: Oggetto

Parametro	Oggetto
Formato	
Categorie	Lunghezza
Lunghezza [O_LNG]	1 Vuoto [error] 2 1<=[O_LNG]<=30 [O_LNG_OK] 3 [O_LNG]>30 [error]

Parametro: Corpo

Parametro	Corpo
Formato	
Categorie	Lunghezza
Lunghezza [C_LNG]	1 [C_LNG]<=700[C_LNG_OK] 2 [C_LNG]>700 [error]

In base a questa scelta delle categorie, le possibili combinazioni sono:

O_LNG	C_LNG	Oracolo
1	1	error
1	2	error
2	1	O_LNG_OK, C_LNG_OK
2	2	error
3	1	error
3	2	error

I test case generati sono quindi:

Test Case ID: TC 2.1					
Precondizioni:					
L'utente si trova sulla pagina che consente di inviare un messaggio					
Flusso di Eventi:					
<ul style="list-style-type: none"> L'utente inserisce i seguenti dati: <table border="1"> <tr> <td>Oggetto</td><td></td></tr> <tr> <td>Corpo</td><td>Corpo Prova</td></tr> </table> L'utente preme sul pulsante "Invio Messaggio" 		Oggetto		Corpo	Corpo Prova
Oggetto					
Corpo	Corpo Prova				
Oracolo:					
L'invio del messaggio non è avvenuto in quanto l'oggetto è vuoto					

Test Case ID: TC 2.2					
Precondizioni:					
L'utente si trova sulla pagina che consente di inviare un messaggio					
Flusso di Eventi:					
<ul style="list-style-type: none"> L'utente inserisce i seguenti dati: <table border="1"> <tr> <td>Oggetto</td><td></td></tr> <tr> <td>Corpo</td><td>//più di 700 caratteri</td></tr> </table> L'utente preme sul pulsante "Invio Messaggio" 		Oggetto		Corpo	//più di 700 caratteri
Oggetto					
Corpo	//più di 700 caratteri				
Oracolo:					
L'invio del messaggio non è avvenuto in quanto il corpo è troppo grande e l'oggetto è vuoto.					

Test Case ID: TC 2.3					
Precondizioni:					
L'utente si trova sulla pagina che consente di inviare un messaggio					
Flusso di Eventi:					
<ul style="list-style-type: none"> L'utente inserisce i seguenti dati: <table border="1"> <tr> <td>Oggetto</td><td>Prova Oggetto</td></tr> <tr> <td>Corpo</td><td>Corpo Esempio</td></tr> </table> L'utente preme sul pulsante "Invio Messaggio" 		Oggetto	Prova Oggetto	Corpo	Corpo Esempio
Oggetto	Prova Oggetto				
Corpo	Corpo Esempio				
Oracolo:					
L'invio del messaggio è avvenuto con successo.					

Test Case ID: TC 2.4					
Precondizioni:					
L'utente si trova sulla pagina che consente di inviare un messaggio					
Flusso di Eventi:					
<ul style="list-style-type: none"> L'utente inserisce i seguenti dati: <table border="1"> <tr> <td>Oggetto</td><td>Prova Oggetto</td></tr> <tr> <td>Corpo</td><td>//più di 700</td></tr> </table> L'utente preme sul pulsante "Invio Messaggio" 		Oggetto	Prova Oggetto	Corpo	//più di 700
Oggetto	Prova Oggetto				
Corpo	//più di 700				
Oracolo:					
L'invio del messaggio non è avvenuto in quanto il corpo supera il numero massimo di caratteri consentito.					

Test Case ID: TC 2.5					
Precondizioni:					
L'utente si trova sulla pagina che consente di inviare un messaggio					
Flusso di Eventi:					
<ul style="list-style-type: none"> L'utente inserisce i seguenti dati: <table border="1"> <tr> <td>Oggetto</td><td>Informazioni Testing Eccessivamente Lunghe Davvero Lunghe</td></tr> <tr> <td>Corpo</td><td>Corpo Esempio</td></tr> </table> L'utente preme sul pulsante "Invio Messaggio" 		Oggetto	Informazioni Testing Eccessivamente Lunghe Davvero Lunghe	Corpo	Corpo Esempio
Oggetto	Informazioni Testing Eccessivamente Lunghe Davvero Lunghe				
Corpo	Corpo Esempio				
Oracolo:					
L'invio del messaggio non è avvenuto in quanto l'oggetto supera il limite massimo di caratteri consentiti.					

Test Case ID: TC 2.6					
Precondizioni:					
L'utente si trova sulla pagina che consente di inviare un messaggio					
Flusso di Eventi:					
<ul style="list-style-type: none"> L'utente inserisce i seguenti dati: <table border="1"> <tr> <td>Oggetto</td><td>Informazioni Testing Eccessivamente Lunghe Davvero Lunghe</td></tr> <tr> <td>Corpo</td><td>//più di 700</td></tr> </table> L'utente preme sul pulsante "Invio Messaggio" 		Oggetto	Informazioni Testing Eccessivamente Lunghe Davvero Lunghe	Corpo	//più di 700
Oggetto	Informazioni Testing Eccessivamente Lunghe Davvero Lunghe				
Corpo	//più di 700				
Oracolo:					
L'invio del messaggio non è avvenuto in quanto l'oggetto è troppo lungo.					

3.1 Salva Setup

Scelta delle Categorie:

Parametro: Carico Aerodinamico Anteriore

Parametro	Carico Aerodinamico Anteriore
Formato	$1 \leq caa \leq 10$
Categorie	Valore
Valore [VALUE]	1 Valore in-Range [CAA_OK] 2 Valore out of Range [error]

Parametro: Carico Aerodinamico Posteriore

Parametro	Carico Aerodinamico Posteriore
Formato	$1 \leq cap \leq 10$
Categorie	Valore
Valore[VALUE]	1 Valore in-Range [if CAA_OK] [CAP_OK] 2 Valore out of Range [error]

Parametro: Campanatura Anteriore

Parametro	Campanatura Anteriore
Formato	$-5 \leq ca \leq 5$
Categorie	Valore
Valore[VALUE]	1 Valore in-Range [if CAP_OK] [CAMPANT_OK] 2 Valore out of Range [error]

Parametro: Campanatura Posteriore

Parametro	Campanatura Posteriore
Formato	$-5 \leq cp \leq 5$
Categorie	Valore
Valore[VALUE]	1 Valore in-Range [if CAMPANT_OK] [CAMPOST_OK] 2 Valore out of Range [error]

Parametro: Convergenza Anteriore

Parametro	Convergenza Posteriore
-----------	------------------------

Formato	-1<=convA<=1
Categorie	Valore
Valore[VALUE]	1 Valore in-Range [if CAMPPOST_OK] [CONVANT_OK] 2 Valore out of Range [error]

Parametro: Convergenza Posteriore

Parametro	Convergenza Posteriore
Formato	-1<=convP<=1
Categorie	Valore
Valore[VALUE]	1 Valore in-Range [if CONVANT_OK] [CONVPOST_OK] 2 Valore out of Range [error]

Parametro: Pressione dei Freni

Parametro	Carico Aerodinamico Posteriore
Formato	0<=PF<=100
Categorie	Valore
Valore[VALUE]	1 Valore in-Range [if CONVPOST_OK] [PRESSF_OK] 2 Valore out of Range [error]

Parametro: Barra Antirollio Posteriore

Parametro	Carico Aerodinamico Posteriore
Formato	1<=BAP<=10
Categorie	Valore
Valore[VALUE]	1 Valore in-Range [if PRESSF_OK] [BAP_OK] 2 Valore out of Range [error]

Parametro: Barra Antirollio Anteriore

Parametro	Carico Aerodinamico Posteriore
Formato	1<=BAA<=10

Categorie	Valore
Valore[VALUE]	1 Valore in-Range [if BAP_OK] [BAA_OK] 2 Valore out of Range [error]

Abbiamo considerato di utilizzare la tecnica BVA, o comunque di considerare i comportamenti agli estremi degli intervalli, ma va tenuto conto del fatto che questa funzionalità si limita semplicemente a salvare un Setup con certi parametri, che possono essere in-range oppure out of range, e non è una funzionalità che prende dei parametri ed effettua dei calcoli.

Funzionalità che utilizzano in qualche modo i parametri inseriti effettuando computazioni necessitano di un'analisi più approfondita agli estremi dell'intervallo.

Vanno inoltre considerate delle eventualità legate all'ambiente, in quanto ogni Setup è specificatamente legato alla presenza nel database di un Circuito a esso associato.

Di conseguenza, se il circuito selezionato viene rimosso dal DB, si incappano in particolari problematiche. Indicheremo quindi come parametro dell'ambiente:

Parametro dell'ambiente: Circuito

Parametro	Circuito
Formato	//
Categorie	Esistenza
Valore[EST]	1 Esistente nel Database [if BAA_OK] [EST_OK] 2 Non esistente nel Database [error]

Le combinazioni sono quindi:

CAA	CAP	CAMP ANT	CAMP POST	CONV ANT	CONV POST	PRES SF	BAP	BAA	Circuit o	Oracolo
2	//	//	//	//	//	//	//	//	//	error
1	2	//	//	//	//	//	//	//	//	error
1	1	2	//	//	//	//	//	//	//	error
1	1	1	2							error
1	1	1	1	2						error
1	1	1	1	1	2					error
1	1	1	1	1	1	2				error
1	1	1	1	1	1	1	2			error
1	1	1	1	1	1	1	1	2		error
1	1	1	1	1	1	1	1	1	2	error
1	1	1	1	1	1	1	1	1	1	[CAA_OK] [CAP_OK] [CAMPANT_ OK] [CAMPPOST_ OK] [CONVANT_

										OK] [CONVPOST_ OK] [PRESSF_OK] [BAP_OK] [BAA_OK] [EST_OK]
--	--	--	--	--	--	--	--	--	--	--

I test case generati sono:

Test Case ID: TC 3.1																			
Precondizioni:																			
L'utente si trova sulla pagina che consente di salvare un Setup																			
Flusso di Eventi:																			
<ul style="list-style-type: none"> L'utente inserisce i seguenti dati: <table border="1"> <tr> <td>Carico Aerodinamico Anteriore</td><td>16</td></tr> <tr> <td>Carico Aerodinamico Posteriore</td><td>5</td></tr> <tr> <td>Campanatura Anteriore</td><td>5</td></tr> <tr> <td>Campanatura Posteriore</td><td>5</td></tr> <tr> <td>Convergenza Anteriore</td><td>1</td></tr> <tr> <td>Convergenza Posteriore</td><td>1</td></tr> <tr> <td>Pressione dei Freni</td><td>43</td></tr> <tr> <td>Barra Antirollio Posteriore</td><td>6</td></tr> <tr> <td>Barra Antirollio Anteriore</td><td>7</td></tr> </table> L'utente preme sul pulsante "Salva Setup" 		Carico Aerodinamico Anteriore	16	Carico Aerodinamico Posteriore	5	Campanatura Anteriore	5	Campanatura Posteriore	5	Convergenza Anteriore	1	Convergenza Posteriore	1	Pressione dei Freni	43	Barra Antirollio Posteriore	6	Barra Antirollio Anteriore	7
Carico Aerodinamico Anteriore	16																		
Carico Aerodinamico Posteriore	5																		
Campanatura Anteriore	5																		
Campanatura Posteriore	5																		
Convergenza Anteriore	1																		
Convergenza Posteriore	1																		
Pressione dei Freni	43																		
Barra Antirollio Posteriore	6																		
Barra Antirollio Anteriore	7																		
Oggetto dell'Ambiente: Circuito di SPA, ID=1, selezionato in precedenza																			
Oracolo:																			
Il setup non viene salvato in quanto il Carico Aerodinamico Anteriore è fuori dal range.																			

Test Case ID: TC 3.2**Precondizioni:**

L'utente si trova sulla pagina che consente di salvare un Setup

Flusso di Eventi:

- L'utente inserisce i seguenti dati:

Carico Aerodinamico Anteriore	8
Carico Aerodinamico Posteriore	15
Campanatura Anteriore	5
Campanatura Posteriore	5
Convergenza Anteriore	1
Convergenza Posteriore	1
Pressione dei Freni	43
Barra Antirollio Posteriore	6
Barra Antirollio Anteriore	7

- L'utente preme sul pulsante "Salva Setup"

Oggetto dell'Ambiente: Circuito di SPA, ID=1, selezionato in precedenza

Oracolo:

Il setup non viene salvato in quanto il Carico Aerodinamico Posteriore è fuori dal range.

Test Case ID: TC 3.3																			
Precondizioni:																			
L'utente si trova sulla pagina che consente di salvare un Setup																			
Flusso di Eventi:																			
<ul style="list-style-type: none"> L'utente inserisce i seguenti dati: <table border="1"> <tr> <td>Carico Aerodinamico Anteriore</td><td>8</td></tr> <tr> <td>Carico Aerodinamico Posteriore</td><td>6</td></tr> <tr> <td>Campanatura Anteriore</td><td>7</td></tr> <tr> <td>Campanatura Posteriore</td><td>5</td></tr> <tr> <td>Convergenza Anteriore</td><td>1</td></tr> <tr> <td>Convergenza Posteriore</td><td>1</td></tr> <tr> <td>Pressione dei Freni</td><td>43</td></tr> <tr> <td>Barra Antirollio Posteriore</td><td>6</td></tr> <tr> <td>Barra Antirollio Anteriore</td><td>7</td></tr> </table> L'utente preme sul pulsante "Salva Setup" 		Carico Aerodinamico Anteriore	8	Carico Aerodinamico Posteriore	6	Campanatura Anteriore	7	Campanatura Posteriore	5	Convergenza Anteriore	1	Convergenza Posteriore	1	Pressione dei Freni	43	Barra Antirollio Posteriore	6	Barra Antirollio Anteriore	7
Carico Aerodinamico Anteriore	8																		
Carico Aerodinamico Posteriore	6																		
Campanatura Anteriore	7																		
Campanatura Posteriore	5																		
Convergenza Anteriore	1																		
Convergenza Posteriore	1																		
Pressione dei Freni	43																		
Barra Antirollio Posteriore	6																		
Barra Antirollio Anteriore	7																		
Oggetto dell'Ambiente: Circuito di SPA, ID=1, selezionato in precedenza																			
Oracolo:																			
Il setup non viene salvato in quanto Campanatura Anteriore è fuori dal range.																			

Test Case ID: TC 3.4**Precondizioni:**

L'utente si trova sulla pagina che consente di salvare un Setup

Flusso di Eventi:

- L'utente inserisce i seguenti dati:

Carico Aerodinamico Anteriore	8
Carico Aerodinamico Posteriore	6
Campanatura Anteriore	5
Campanatura Posteriore	7
Convergenza Anteriore	1
Convergenza Posteriore	1
Pressione dei Freni	43
Barra Antirollio Posteriore	6
Barra Antirollio Anteriore	7

- L'utente preme sul pulsante "Salva Setup"

Oggetto dell'Ambiente: Circuito di SPA, ID=1, selezionato in precedenza

Oracolo:

Il setup non viene salvato in quanto Campanatura Posteriore è fuori dal range.

Test Case ID: TC 3.5**Precondizioni:**

L'utente si trova sulla pagina che consente di salvare un Setup

Flusso di Eventi:

- L'utente inserisce i seguenti dati:

Carico Aerodinamico Anteriore	8
Carico Aerodinamico Posteriore	6
Campanatura Anteriore	5
Campanatura Posteriore	3
Convergenza Anteriore	7
Convergenza Posteriore	1
Pressione dei Freni	43
Barra Antirollio Posteriore	6
Barra Antirollio Anteriore	7

- L'utente preme sul pulsante "Salva Setup"

Oggetto dell'Ambiente: Circuito di SPA, ID=1, selezionato in precedenza

Oracolo:

Il setup non viene salvato in quanto Convergenza Anteriore è fuori dal range.

Test Case ID: TC 3.6

Precondizioni:

L'utente si trova sulla pagina che consente di salvare un Setup

Flusso di Eventi:

- L'utente inserisce i seguenti dati:

Carico Aerodinamico Anteriore	8
Carico Aerodinamico Posteriore	6
Campanatura Anteriore	5
Campanatura Posteriore	3
Convergenza Anteriore	1
Convergenza Posteriore	7
Pressione dei Freni	43
Barra Antirollio Posteriore	6
Barra Antirollio Anteriore	7

- L'utente preme sul pulsante "Salva Setup"

Oggetto dell'Ambiente: Circuito di SPA, ID=1, selezionato in precedenza

Oracolo:

Il setup non viene salvato in quanto Convergenza Posteriore è fuori dal range.

Test Case ID: TC 3.7																			
Precondizioni:																			
L'utente si trova sulla pagina che consente di salvare un Setup																			
Flusso di Eventi:																			
<ul style="list-style-type: none">L'utente inserisce i seguenti dati:<table><tr><td>Carico Aerodinamico Anteriore</td><td>8</td></tr><tr><td>Carico Aerodinamico Posteriore</td><td>6</td></tr><tr><td>Campanatura Anteriore</td><td>5</td></tr><tr><td>Campanatura Posteriore</td><td>3</td></tr><tr><td>Convergenza Anteriore</td><td>1</td></tr><tr><td>Convergenza Posteriore</td><td>1</td></tr><tr><td>Pressione dei Freni</td><td>167</td></tr><tr><td>Barra Antirollio Posteriore</td><td>6</td></tr><tr><td>Barra Antirollio Anteriore</td><td>7</td></tr></table>L'utente preme sul pulsante "Salva Setup"		Carico Aerodinamico Anteriore	8	Carico Aerodinamico Posteriore	6	Campanatura Anteriore	5	Campanatura Posteriore	3	Convergenza Anteriore	1	Convergenza Posteriore	1	Pressione dei Freni	167	Barra Antirollio Posteriore	6	Barra Antirollio Anteriore	7
Carico Aerodinamico Anteriore	8																		
Carico Aerodinamico Posteriore	6																		
Campanatura Anteriore	5																		
Campanatura Posteriore	3																		
Convergenza Anteriore	1																		
Convergenza Posteriore	1																		
Pressione dei Freni	167																		
Barra Antirollio Posteriore	6																		
Barra Antirollio Anteriore	7																		
Oggetto dell'Ambiente: Circuito di SPA, ID=1, selezionato in precedenza																			
Oracolo:																			
Il setup non viene salvato in quanto Pressione dei Freni è fuori dal range.																			

Test Case ID: TC 3.8

Precondizioni:

L'utente si trova sulla pagina che consente di salvare un Setup

Flusso di Eventi:

- L'utente inserisce i seguenti dati:

Carico Aerodinamico Anteriore	8
Carico Aerodinamico Posteriore	6
Campanatura Anteriore	5
Campanatura Posteriore	3
Convergenza Anteriore	1
Convergenza Posteriore	1
Pressione dei Freni	80
Barra Antirollio Posteriore	15
Barra Antirollio Anteriore	7

- L'utente preme sul pulsante "Salva Setup"

Oggetto dell'Ambiente: Circuito di SPA, ID=1, selezionato in precedenza

Oracolo:

Il setup non viene salvato in quanto Barra Antirollio Posteriore è fuori dal range.

Test Case ID: TC 3.9**Precondizioni:**

L'utente si trova sulla pagina che consente di salvare un Setup

Flusso di Eventi:

- L'utente inserisce i seguenti dati:

Carico Aerodinamico Anteriore	8
Carico Aerodinamico Posteriore	6
Campanatura Anteriore	5
Campanatura Posteriore	3
Convergenza Anteriore	1
Convergenza Posteriore	1
Pressione dei Freni	80
Barra Antirollio Posteriore	7
Barra Antirollio Anteriore	16

- L'utente preme sul pulsante "Salva Setup"

Oggetto dell'Ambiente: Circuito di SPA, ID=1, selezionato in precedenza

Oracolo:

Il setup non viene salvato in quanto Barra Antirollio Anteriore è fuori dal range.

Test Case ID: TC 3.10**Precondizioni:**

L'utente si trova sulla pagina che consente di salvare un Setup

Flusso di Eventi:

- L'utente inserisce i seguenti dati:

Carico Aerodinamico Anteriore	8
Carico Aerodinamico Posteriore	6
Campanatura Anteriore	5
Campanatura Posteriore	3
Convergenza Anteriore	1
Convergenza Posteriore	1
Pressione dei Freni	80
Barra Antirollio Posteriore	7
Barra Antirollio Anteriore	7

- L'utente preme sul pulsante "Salva Setup"

Oggetto dell'Ambiente: Circuito di Pontecagnano Faiano, ID=13543, selezionato in precedenza

Oracolo:

Il setup non viene salvato in quanto il circuito di Pontecagnano Faiano con ID=13543 non esiste nel DB.

Test Case ID: TC 3.11																			
Precondizioni:																			
L'utente si trova sulla pagina che consente di salvare un Setup																			
Flusso di Eventi:																			
<ul style="list-style-type: none"> L'utente inserisce i seguenti dati: <table border="1"> <tr> <td>Carico Aerodinamico Anteriore</td><td>8</td></tr> <tr> <td>Carico Aerodinamico Posteriore</td><td>6</td></tr> <tr> <td>Campanatura Anteriore</td><td>5</td></tr> <tr> <td>Campanatura Posteriore</td><td>3</td></tr> <tr> <td>Convergenza Anteriore</td><td>0</td></tr> <tr> <td>Convergenza Posteriore</td><td>1</td></tr> <tr> <td>Pressione dei Freni</td><td>80</td></tr> <tr> <td>Barra Antirollio Posteriore</td><td>7</td></tr> <tr> <td>Barra Antirollio Anteriore</td><td>7</td></tr> </table> L'utente preme sul pulsante "Salva Setup" 		Carico Aerodinamico Anteriore	8	Carico Aerodinamico Posteriore	6	Campanatura Anteriore	5	Campanatura Posteriore	3	Convergenza Anteriore	0	Convergenza Posteriore	1	Pressione dei Freni	80	Barra Antirollio Posteriore	7	Barra Antirollio Anteriore	7
Carico Aerodinamico Anteriore	8																		
Carico Aerodinamico Posteriore	6																		
Campanatura Anteriore	5																		
Campanatura Posteriore	3																		
Convergenza Anteriore	0																		
Convergenza Posteriore	1																		
Pressione dei Freni	80																		
Barra Antirollio Posteriore	7																		
Barra Antirollio Anteriore	7																		
Oggetto dell'Ambiente: Circuito di SPA, ID=1, selezionato in precedenza																			
Oracolo:																			
Il setup viene salvato correttamente.																			

4.1 Modifica Setup

Scelta delle Categorie:

Parametro: ID

Parametro	ID
Formato	IdSetup>0
Categorie	Esistenza
Esistenza [IDSETUP_EST]	<ul style="list-style-type: none">1 Non esiste nel database [error]2 Esiste nel database [IDSETUP_EST_OK]

Parametro: Carico Aerodinamico Anteriore

Parametro	Carico Aerodinamico Anteriore
Formato	1<=caa<=10
Categorie	Valore
Valore [CAA_VLR]	<ul style="list-style-type: none">1.1 Valore in-Range [if IDSETUP_EST_OK] [CAA_OK]1.2 Valore out of Range [if IDSETUP_EST_OK] [error]

Parametro: Carico Aerodinamico Posteriore

Parametro	Carico Aerodinamico Posteriore
Formato	1<=cap<=10
Categorie	Valore
Valore [CAP_VLR]	<ul style="list-style-type: none">2.1 Valore in-Range [if CAA_OK][CAP_OK]2.2 Valore out of Range [if CAA_OK][[error]

Parametro: Campanatura Anteriore

Parametro	Campanatura Anteriore
Formato	-5<=ca<=5
Categorie	Valore
Valore [CAMPANT_VLR]	<ul style="list-style-type: none">2.1 Valore in-Range [if CAP_OK] [CAMPANT_OK]2.2 Valore out of Range [if CAP_OK][[error]

Parametro: Campanatura Posteriore

Parametro	Campanatura Posteriore
Formato	-5<=cp<=5
Categorie	Valore
Valore[CAMPPOST_VLR]	2.1 Valore in-Range [if CAMPANT_OK] [CAMPPOST_OK] 2.2 Valore out of Range [if CAMPANT_OK] [[error]

Parametro: Convergenza Anteriore

Parametro	Convergenza Posteriore
Formato	-1<=convA<=1
Categorie	Valore
Valore[CONVANT_VLR]	2.1 Valore in-Range [if CAMPPOST_OK] [CONVANT_OK] 2.2 Valore out of Range[if CAMPPOST_OK] [[error]

Parametro: Convergenza Posteriore

Parametro	Convergenza Posteriore
Formato	-1<=convP<=1
Categorie	Valore
Valore[CONVPOST_VLR]	2.1 Valore in-Range [if CONVANT_OK] [CONVPOST_OK] 2.2 Valore out of Range[if CONVANT_OK] [[error]

Parametro: Pressione dei Freni

Parametro	Carico Aerodinamico Posteriore
Formato	0<=PF<=100
Categorie	Valore
Valore[PRESSF_VLR]	2.1 Valore in-Range [if CONVPOST_OK] [PRESSF_OK] 2.2 Valore out of Range [if CONVPOST_OK][[error]

Parametro: Barra Antirollio Posteriore

Parametro	Carico Aerodinamico Posteriore
Formato	1<=BAP<=10
Categorie	Valore
Valore[BAP_VLR]	2.1 Valore in-Range [if PRESSF_OK] [BAP_OK] 2.2 Valore out of Range[if PRESSF_OK] [[error]

Parametro: Barra Antirollio Anteriore

Parametro	Carico Aerodinamico Posteriore
Formato	1<=BAA<=10
Categorie	Valore
Valore[BAA_VLR]	2.1 Valore in-Range [if BAP_OK][BAA_OK] 2.2 Valore out of Range [if BAP_OK][[error]

ID	CAA	CAP	CAMP ANT	CAMP POST	CON VANT	CONV POST	PRESSF	BAP	BAA	Oracolo
1	\\	\\	\\	\\	\\	\\	\\	\\	\\	Error
2	2	\\	\\	\\	\\	\\	\\	\\	\\	Error
2	1	2	\\	\\	\\	\\	\\	\\	\\	Error
2	1	1	2	\\	\\	\\	\\	\\	\\	Error
2	1	1	1	2	\\	\\	\\	\\	\\	Error
2	1	1	1	1	2	\\	\\	\\	\\	Error
2	1	1	1	1	1	2	\\	\\	\\	Error
2	1	1	1	1	1	1	2	\\	\\	Error
2	1	1	1	1	1	1	1	2	\\	Error
2	1	1	1	1	1	1	1	1	2	Error
2	1	1	1	1	1	1	1	1	1	[CAA_OK] [CAP_OK] [CAMPANT_OK] [CAMPPOST_OK]

										[CONVANT_OK] [CONVPOST_OK] [PRESSF_OK] [BAP_OK] [BAA_OK] [IDSETUP_EST _OK]
--	--	--	--	--	--	--	--	--	--	---

I test case generati sono:

Test Case ID: TC 4.1																			
Precondizioni:																			
L'utente si trova sulla pagina che consente di modificare un Setup e cerca di modificare un Setup con id=-1																			
Flusso di Eventi:																			
<ul style="list-style-type: none"> L'utente inserisce i seguenti dati: <table> <tr> <td>Carico Aerodinamico Anteriore</td><td>8</td></tr> <tr> <td>Carico Aerodinamico Posteriore</td><td>6</td></tr> <tr> <td>Campanatura Anteriore</td><td>5</td></tr> <tr> <td>Campanatura Posteriore</td><td>3</td></tr> <tr> <td>Convergenza Anteriore</td><td>4</td></tr> <tr> <td>Convergenza Posteriore</td><td>4</td></tr> <tr> <td>Pressione dei Freni</td><td>80</td></tr> <tr> <td>Barra Antirollio Posteriore</td><td>7</td></tr> <tr> <td>Barra Antirollio Anteriore</td><td>7</td></tr> </table> L'utente preme sul pulsante "Salva" 		Carico Aerodinamico Anteriore	8	Carico Aerodinamico Posteriore	6	Campanatura Anteriore	5	Campanatura Posteriore	3	Convergenza Anteriore	4	Convergenza Posteriore	4	Pressione dei Freni	80	Barra Antirollio Posteriore	7	Barra Antirollio Anteriore	7
Carico Aerodinamico Anteriore	8																		
Carico Aerodinamico Posteriore	6																		
Campanatura Anteriore	5																		
Campanatura Posteriore	3																		
Convergenza Anteriore	4																		
Convergenza Posteriore	4																		
Pressione dei Freni	80																		
Barra Antirollio Posteriore	7																		
Barra Antirollio Anteriore	7																		
Oggetto dell'Ambiente: //																			
Oracolo:																			
Il setup non viene modificato in quanto non è presente alcun setup con id=-1 nel DataBase.																			

Test Case ID: TC 4.2**Precondizioni:**

L'utente si trova sulla pagina che consente di modificare un Setup

Flusso di Eventi:

- L'utente inserisce i seguenti dati:

Carico Aerodinamico Anteriore	16
Carico Aerodinamico Posteriore	5
Campanatura Anteriore	5
Campanatura Posteriore	5
Convergenza Anteriore	1
Convergenza Posteriore	1
Pressione dei Freni	43
Barra Antirollio Posteriore	6
Barra Antirollio Anteriore	7

- L'utente preme sul pulsante "Salva"

Oggetto dell'Ambiente: //

Oracolo:

Il setup non viene modificato in quanto il Carico Aerodinamico Anteriore è fuori dal range.

Test Case ID: TC 4.3**Precondizioni:**

L'utente si trova sulla pagina che consente di modificare un Setup

Flusso di Eventi:

- L'utente inserisce i seguenti dati:

Carico Aerodinamico Anteriore	8
Carico Aerodinamico Posteriore	15
Campanatura Anteriore	5
Campanatura Posteriore	5
Convergenza Anteriore	1
Convergenza Posteriore	1
Pressione dei Freni	43
Barra Antirollio Posteriore	6
Barra Antirollio Anteriore	7

- L'utente preme sul pulsante "Salva"

Oggetto dell'Ambiente: //

Oracolo:

Il setup non viene modificato in quanto il Carico Aerodinamico Posteriore è fuori dal range.

Test Case ID: TC 4.4																			
Precondizioni:																			
L'utente si trova sulla pagina che consente di modificare un Setup																			
Flusso di Eventi:																			
<ul style="list-style-type: none"> L'utente inserisce i seguenti dati: <table border="1"> <tr> <td>Carico Aerodinamico Anteriore</td><td>8</td></tr> <tr> <td>Carico Aerodinamico Posteriore</td><td>6</td></tr> <tr> <td>Campanatura Anteriore</td><td>7</td></tr> <tr> <td>Campanatura Posteriore</td><td>5</td></tr> <tr> <td>Convergenza Anteriore</td><td>1</td></tr> <tr> <td>Convergenza Posteriore</td><td>1</td></tr> <tr> <td>Pressione dei Freni</td><td>43</td></tr> <tr> <td>Barra Antirollio Posteriore</td><td>6</td></tr> <tr> <td>Barra Antirollio Anteriore</td><td>7</td></tr> </table> L'utente preme sul pulsante "Salva" 		Carico Aerodinamico Anteriore	8	Carico Aerodinamico Posteriore	6	Campanatura Anteriore	7	Campanatura Posteriore	5	Convergenza Anteriore	1	Convergenza Posteriore	1	Pressione dei Freni	43	Barra Antirollio Posteriore	6	Barra Antirollio Anteriore	7
Carico Aerodinamico Anteriore	8																		
Carico Aerodinamico Posteriore	6																		
Campanatura Anteriore	7																		
Campanatura Posteriore	5																		
Convergenza Anteriore	1																		
Convergenza Posteriore	1																		
Pressione dei Freni	43																		
Barra Antirollio Posteriore	6																		
Barra Antirollio Anteriore	7																		
Oggetto dell'Ambiente: //																			
Oracolo:																			
Il setup non viene modificato in quanto Campanatura Anteriore è fuori dal range.																			

Test Case ID: TC 4.5**Precondizioni:**

L'utente si trova sulla pagina che consente di modificare un Setup

Flusso di Eventi:

- L'utente inserisce i seguenti dati:

Carico Aerodinamico Anteriore	8
Carico Aerodinamico Posteriore	6
Campanatura Anteriore	5
Campanatura Posteriore	7
Convergenza Anteriore	1
Convergenza Posteriore	1
Pressione dei Freni	43
Barra Antirollio Posteriore	6
Barra Antirollio Anteriore	7

- L'utente preme sul pulsante "Setup"

Oggetto dell'Ambiente: //

Oracolo:

Il setup non viene modificato in quanto Campanatura Posteriore è fuori dal range.

Test Case ID: TC 4.6																			
Precondizioni:																			
L'utente si trova sulla pagina che consente di modificare un Setup																			
Flusso di Eventi:																			
<ul style="list-style-type: none"> L'utente inserisce i seguenti dati: <table border="1"> <tr> <td>Carico Aerodinamico Anteriore</td><td>8</td></tr> <tr> <td>Carico Aerodinamico Posteriore</td><td>6</td></tr> <tr> <td>Campanatura Anteriore</td><td>5</td></tr> <tr> <td>Campanatura Posteriore</td><td>3</td></tr> <tr> <td>Convergenza Anteriore</td><td>7</td></tr> <tr> <td>Convergenza Posteriore</td><td>4</td></tr> <tr> <td>Pressione dei Freni</td><td>43</td></tr> <tr> <td>Barra Antirollio Posteriore</td><td>6</td></tr> <tr> <td>Barra Antirollio Anteriore</td><td>7</td></tr> </table> L'utente preme sul pulsante "Salva" 		Carico Aerodinamico Anteriore	8	Carico Aerodinamico Posteriore	6	Campanatura Anteriore	5	Campanatura Posteriore	3	Convergenza Anteriore	7	Convergenza Posteriore	4	Pressione dei Freni	43	Barra Antirollio Posteriore	6	Barra Antirollio Anteriore	7
Carico Aerodinamico Anteriore	8																		
Carico Aerodinamico Posteriore	6																		
Campanatura Anteriore	5																		
Campanatura Posteriore	3																		
Convergenza Anteriore	7																		
Convergenza Posteriore	4																		
Pressione dei Freni	43																		
Barra Antirollio Posteriore	6																		
Barra Antirollio Anteriore	7																		
Oggetto dell'Ambiente://																			
Oracolo:																			
Il setup non viene modificato in quanto Convergenza Anteriore è fuori dal range.																			

Test Case ID: TC 4.7**Precondizioni:**

L'utente si trova sulla pagina che consente di modificare un Setup

Flusso di Eventi:

- L'utente inserisce i seguenti dati:

Carico Aerodinamico Anteriore	8
Carico Aerodinamico Posteriore	6
Campanatura Anteriore	5
Campanatura Posteriore	3
Convergenza Anteriore	1
Convergenza Posteriore	7
Pressione dei Freni	43
Barra Antirollio Posteriore	6
Barra Antirollio Anteriore	7

- L'utente preme sul pulsante "Salva"

Oggetto dell'Ambiente: //

Oracolo:

Il setup non viene modificato in quanto Convergenza Posteriore è fuori dal range.

Test Case ID: TC 4.8**Precondizioni:**

L'utente si trova sulla pagina che consente di modificare un Setup

Flusso di Eventi:

- L'utente inserisce i seguenti dati:

Carico Aerodinamico Anteriore	8
Carico Aerodinamico Posteriore	6
Campanatura Anteriore	5
Campanatura Posteriore	3
Convergenza Anteriore	1
Convergenza Posteriore	1
Pressione dei Freni	167
Barra Antirollio Posteriore	6
Barra Antirollio Anteriore	7

- L'utente preme sul pulsante "Salva"

Oggetto dell'Ambiente: //

Oracolo:

Il setup non viene modificato in quanto Pressione dei Freni è fuori dal range.

Test Case ID: TC 4.9																			
Precondizioni:																			
L'utente si trova sulla pagina che consente di modificare un Setup																			
Flusso di Eventi:																			
<ul style="list-style-type: none"> L'utente inserisce i seguenti dati: <table border="1"> <tr> <td>Carico Aerodinamico Anteriore</td><td>8</td></tr> <tr> <td>Carico Aerodinamico Posteriore</td><td>6</td></tr> <tr> <td>Campanatura Anteriore</td><td>5</td></tr> <tr> <td>Campanatura Posteriore</td><td>3</td></tr> <tr> <td>Convergenza Anteriore</td><td>1</td></tr> <tr> <td>Convergenza Posteriore</td><td>1</td></tr> <tr> <td>Pressione dei Freni</td><td>80</td></tr> <tr> <td>Barra Antirollio Posteriore</td><td>15</td></tr> <tr> <td>Barra Antirollio Anteriore</td><td>7</td></tr> </table> L'utente preme sul pulsante "Salva" 		Carico Aerodinamico Anteriore	8	Carico Aerodinamico Posteriore	6	Campanatura Anteriore	5	Campanatura Posteriore	3	Convergenza Anteriore	1	Convergenza Posteriore	1	Pressione dei Freni	80	Barra Antirollio Posteriore	15	Barra Antirollio Anteriore	7
Carico Aerodinamico Anteriore	8																		
Carico Aerodinamico Posteriore	6																		
Campanatura Anteriore	5																		
Campanatura Posteriore	3																		
Convergenza Anteriore	1																		
Convergenza Posteriore	1																		
Pressione dei Freni	80																		
Barra Antirollio Posteriore	15																		
Barra Antirollio Anteriore	7																		
Oggetto dell'Ambiente://																			
Oracolo:																			
Il setup non viene modificato in quanto Barra Antirollio Posteriore è fuori dal range.																			

Test Case ID: TC 4.10**Precondizioni:**

L'utente si trova sulla pagina che consente di modificare un Setup

Flusso di Eventi:

- L'utente inserisce i seguenti dati:

Carico Aerodinamico Anteriore	8
Carico Aerodinamico Posteriore	6
Campanatura Anteriore	5
Campanatura Posteriore	3
Convergenza Anteriore	1
Convergenza Posteriore	1
Pressione dei Freni	80
Barra Antirollio Posteriore	7
Barra Antirollio Anteriore	16

- L'utente preme sul pulsante "Salva"

Oggetto dell'Ambiente://

Oracolo:

Il setup non viene modificato in quanto Barra Antirollio Anteriore è fuori dal range.

Test Case ID: TC 4.11**Precondizioni:**

L'utente si trova sulla pagina che consente di modificare un Setup

Flusso di Eventi:

- L'utente inserisce i seguenti dati:

Carico Aerodinamico Anteriore	8
Carico Aerodinamico Posteriore	6
Campanatura Anteriore	5
Campanatura Posteriore	3
Convergenza Anteriore	1
Convergenza Posteriore	1
Pressione dei Freni	80
Barra Antirollio Posteriore	7
Barra Antirollio Anteriore	7

- L'utente preme sul pulsante "Salva"

Oggetto dell'Ambiente: //

Oracolo:

Il setup viene modificato con successo.

5.1 Elimina Setup

Scelta delle Categorie:

Parametro: ID

Parametro	ID
Formato	IdSetup>0
Categorie	Esistenza
Esistenza [IDSETUP_EST]	<div>1 Non esiste nel database [error]</div> <div>2 Esiste nel database [IDSETUP_EST _OK]</div>

IDSETUP	Oracolo
1	Error
2	[IDSETUP_EST _OK]

I test case generati sono:

Test Case ID: TC 5.1
Precondizioni:
L'utente si trova sulla pagina contenente la lista setup
Flusso di Eventi:
<ul style="list-style-type: none">L'utente clicca sul pulsante "elimina" associato al setup con id=6
Oggetto dell'Ambiente: //
Oracolo:
Il setup viene eliminato con successo

--

Test Case ID: TC 5.2
Precondizioni:
L'utente si trova sulla pagina contenente la lista setup
Flusso di Eventi:
<ul style="list-style-type: none">L'utente clicca sul pulsante "elimina" associato al setup con id=-1
Oggetto dell'Ambiente: //
Oracolo:
Il setup non viene eliminato perché non è presente alcun setup con id=-1 all'interno del DataBase.

6.1 Visualizzazione Circuito

Scelta delle Categorie:

Parametro: ID

Parametro	ID
Formato	Intero positivo
Categorie	Formato, Esistenza
Formato [ID_FRM]	<ul style="list-style-type: none">Non rispetta il formato [error]Rispetta il formato [ID_FRM_OK]
Esistente [EST]	3 Non esiste nel database [if ID_FRM_OK and PW_FRM_OK][error]

	4 Esiste nel database [if ID_FRM_OK and PW_FRM_OK][EST_OK]
--	--

In base a questa scelta delle categorie, le possibili combinazioni sono:

ID_FRM	EST	Oracolo
1	//	error
2	1	error
2	2	ID_FRM_OK, EST_OK

I test case generati sono quindi:

Test Case ID: TC 6.1		
Precondizioni:		
L'utente ha effettuato il login		
Flusso di Eventi:		
3 L'utente richiede la visualizzazione di un Circuito con tale ID		
	ID	
Oracolo:		
La richiesta non ha avuto successo perché i campi obbligatori sono vuoti.		

Test Case ID: TC 6.2		
Precondizioni:		
L'utente ha effettuato il login		
Flusso di Eventi:		
4 L'utente richiede la visualizzazione di un Circuito con tale ID		
	ID	abc
Oracolo:		
La richiesta non ha avuto successo perché il campo ID non rispetta il formato.		

Test Case ID: TC 6.3		
Precondizioni:		
L'utente ha effettuato il login		
Flusso di Eventi:		
5 L'utente richiede la visualizzazione di un Circuito con tale ID		
	ID	-1
Oracolo:		
La richiesta non ha avuto successo perché l'ID non esiste nel DB		

Test Case ID: TC 6.4		
Precondizioni:		
L'utente ha effettuato il login		
Flusso di Eventi:		
6 L'utente richiede la visualizzazione di un Circuito con tale ID		
	ID	5
Oracolo:		
La richiesta ha avuto successo		

7.1 Visualizzazione Setup

Scelta delle Categorie:

Parametro: ID

Parametro	ID
Formato	Intero positivo
Categorie	Formato, Esistenza
Formato [ID_FRM]	<ul style="list-style-type: none">Non rispetta il formato [error]Rispetta il formato [ID_FRM_OK]
Esistente [EST]	<ul style="list-style-type: none">5 Non esiste nel database [if ID_FRM_OK and PW_FRM_OK][error]6 Esiste nel database [if ID_FRM_OK and PW_FRM_OK][EST_OK]

In base a questa scelta delle categorie, le possibili combinazioni sono:

ID_FRM	EST	Oracolo
1	//	error
2	1	error
2	2	ID_FRM_OK, EST_OK

I test case generati sono quindi:

Test Case ID: TC 7.1		
Precondizioni:		
L'utente ha effettuato il login		
Flusso di Eventi:		
7 L'utente richiede la visualizzazione di un Setup con tale ID		
	ID	
Oracolo:		

La richiesta non ha avuto successo perché i campi obbligatori sono vuoti.

Test Case ID: TC 7.2		
Precondizioni:		
L'utente ha effettuato il login		
Flusso di Eventi:		
8 L'utente richiede la visualizzazione di un Setup con tale ID		
<table border="1"><tr><td>ID</td><td>abc</td></tr></table>	ID	abc
ID	abc	
Oracolo:		
La richiesta non ha avuto successo perché il campo ID non rispetta il formato.		

Test Case ID: TC 7.3		
Precondizioni:		
L'utente ha effettuato il login		
Flusso di Eventi:		
9 L'utente richiede la visualizzazione di un Setup con tale ID		
<table border="1"><tr><td>ID</td><td>-1</td></tr></table>	ID	-1
ID	-1	
Oracolo:		

La richiesta non ha avuto successo perché l'ID non esiste nel DB
--

Test Case ID: TC 7.4		
Precondizioni:		
L'utente ha effettuato il login		
Flusso di Eventi:		
10 L'utente richiede la visualizzazione di un Setup con tale ID		
<table border="1"><tr><td>ID</td><td>5</td></tr></table>	ID	5
ID	5	
Oracolo:		
La richiesta ha avuto successo		

Test Cases per Test di Unità

1.1 Rettilineo

Scelta delle Categorie:

Parametro: Nome

Parametro	Nome
Formato	(nome != null)&&(nome != '')
Categorie	Formato
Formato [NOME_FRM]	1. Non rispetta il formato [error] 2. Rispetta il formato [NOME_FRM_OK]

Parametro: Lunghezza

Parametro	Lunghezza
Formato	lun>0
Categorie	Valore
VALORE[LUNGHEZZA_VLR]	1. Valore in-Range [LUNGHEZZA_VLR_OK] 2. Valore out of Range [error]

Test Case Name: TestGetID
Precondizioni:
rettilineo.ID == 1
Flusso di Eventi:
1. Alla variabile exp viene assegnato il valore 1 2. Alla variabile act viene assegnato il valore di ritorno rettilineo.getID() 3. Chiamo il metodo assertEquals passando come parametri le variabili exp e act 1.
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestGetLunghezza
Precondizioni:
rettilineo.lunghezza == 400.0
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Alla variabile exp viene assegnato il valore 400. 2. Alla variabile act viene assegnato il valore di ritorno rettilineo.getLunghezza() 3. Chiamo il metodo assertEquals passando come parametri le variabili exp e act
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestGetNome
Precondizioni:
rettilineo.nome == "rettilineo1"
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Alla variabile exp viene assegnata la stringa "rettilineo1" 2. Alla variabile act viene assegnato il valore di ritorno rettilineo.getNome() 3. Chiamo il metodo assertEquals passando come parametri le variabili exp e act <ol style="list-style-type: none"> 1.
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetName
Precondizioni:
rettilineo.nome == "rettilineo1"
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo rettilineo.setNome(String) passando come parametro la stringa "rettilineo2" Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo rettilineo.getNome() e la stringa "rettineo2" <p>1.</p>
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetID
Precondizioni:
rettilineo.ID == 1
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo rettilineo.setID(int) passando come parametro l'intero 2 Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo rettilineo.getID() e l'intero 2 <p>1.</p>
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetLunghezza
Precondizioni:
rettilineo.lunghezza == 700.0
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo rettilineo.setLunghezza(double) passando come parametro il double 700.0 Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo rettilineo.getLunghezza() e il double 700.0 <p>1.</p>
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestNomeNull
Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo rettilineo.setNome(String) passando come parametro il valore null Chiamo il metodo fail0; <p>1.</p>
Oracolo:
non viene generata alcuna eccezione

Test Case Name: TestNomeEmpty
Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Viene chiamato il metodo rettilineo.setNome(String) passando come parametro il valore "" 2. Chiamo il metodo fail0; <p>1.</p>
Oracolo:
non viene cenerata alcuna eccezione

Test Case Name: TestNLunghezzaRange
Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> 3. Viene chiamato il metodo rettilineo.setLunghezza (double) passando come parametro il valore -700.0 4. Chiamo il metodo fail0; <p>1.</p>
Oracolo:
non viene generata alcuna eccezione

2.1 Curva

Scelta delle Categorie:

Parametro: Nome

Parametro	Nome
Formato	(nome != null)&&(nome != '')
Categorie	Formato
Formato [NOME_FRM]	1. Non rispetta il formato [error] 2. Rispetta il formato [NOME_FRM_OK]

Parametro: Angolo

Parametro	Angolo
Formato	angolo>0
Categorie	Valore
VALORE[ANGOLO_VLR]	1. Valore in-Range [ANGOLO_VLR_OK] 2. Valore out of Range [error]

Test Case Name: TestGetID
Precondizioni:
curva.ID == 1
Flusso di Eventi:
1. Alla variabile exp viene assegnato il valore 1 2. Alla variabile act viene assegnato il valore di ritorno curva.getID() 3. Chiamo il metodo assertEquals passando come parametri le variabili exp e act
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestGetAngolo
Precondizioni:
curva.Angolo == 30.0f
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Alla variabile exp viene assegnato il valore 30.0f 2. Alla variabile act viene assegnato il valore di ritorno curva.getAngolo() 3. Chiamo il metodo assertEquals passando come parametri le variabili exp e act
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestGetNome
Precondizioni:
curva.nome == "curva1"
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Alla variabile exp viene assegnata la stringa "rcurva1" 2. Alla variabile act viene assegnato il valore di ritorno curva.getNome() 3. Chiamo il metodo assertEquals passando come parametri le variabili exp e act <ol style="list-style-type: none"> 1.
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetName
Precondizioni:
curva.nome == "curva1"
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo curva.setNome(String) passando come parametro la stringa "curva2" Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo rettilineo.getNome() e la stringa "curva2" <p>1.</p>
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetID
Precondizioni:
curva.ID == 1
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo curva.setID(int) passando come parametro l'intero 2 Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo curva.getID() e l'intero 2 <p>1.</p>
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetAngolo
Precondizioni:
curva.nome == 30.0f
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo curva.setAngolo(float) passando come parametro il float 23.0f Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo curva.getAngolo() e il float 23.0f <p>1.</p>
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestNomeNull
Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo curva.setNome(String) passando come parametro il valore null Chiamo il metodo fail0; <p>1.</p>
Oracolo:
non viene generata alcuna eccezione

Test Case Name: TestNomeEmpty
Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Viene chiamato il metodo curva.setNome(String) passando come parametro il valore "" 2. Chiamo il metodo fail0; <p>1.</p>
Oracolo:
non viene cenerata alcuna eccezione

Test Case Name: TestAngoloRange
Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> 3. Viene chiamato il metodo curva.setAngolo (float) passando come parametro il valore -50.0f 4. Chiamo il metodo fail0; <p>1.</p>
Oracolo:
non viene cenerata alcuna eccezione

3.1 Circuito

Scelta delle Categorie:

Parametro: Nome

Parametro	Nome
Formato	(nome != null)&&(nome != '')
Categorie	Formato
Formato [NOME_FRM]	<ol style="list-style-type: none">1. Non rispetta il formato [error]2. Rispetta il formato [NOME_FRM_OK]

Parametro: Sede

Parametro	Sede
Formato	(sede != null)&&(sede != '')
Categorie	Formato
Formato [SEDE_FRM]	<ol style="list-style-type: none">1. Non rispetta il formato [error]2. Rispetta il formato [SEDE_FRM_OK]

Parametro: Lunghezza

Parametro	Lunghezza
Formato	lunghezza>0
Categorie	Valore
VALORE[LUNGHEZZA_VLR]	<ol style="list-style-type: none">1. Valore in-Range [LUNGHEZZA_VLR_OK]2. Valore out of Range [error]

Parametro: Meteo

Parametro	Meteo
Formato	(meteo != null)&&(meteo != '')
Categorie	Formato
Formato [METEO_FRM]	<ol style="list-style-type: none">1. Non rispetta il formato [error]2. Rispetta il formato [METEO_FRM_OK]

Parametro: TPM

Parametro	TPM
Formato	tpm>0
Categorie	Valore
VALORE[TPM_VLR]	<ol style="list-style-type: none">1. Valore in-Range [TPM_VLR_OK]2. Valore out of Range [error]

Parametro: RPM

Parametro	RPM
Formato	rpm>0
Categorie	Valore
VALORE[RPM_VLR]	<ol style="list-style-type: none">1. Valore in-Range [RPM_VLR_OK]2. Valore out of Range [error]

Parametro: Umidità

Parametro	Umidità
Formato	(umidità != null)&&(umidità != “)
Categorie	Formato
Formato [UMIDITÀ_FRM]	<ol style="list-style-type: none">1. Non rispetta il formato [error]2. Rispetta il formato [UMIDITÀ_FRM_OK]

Parametro: NumeroGiri

Parametro	NumeroGiri
Formato	numeroGiri>0
Categorie	Valore
VALORE[NGIRI_VLR]	<ol style="list-style-type: none">1. Valore in-Range [NGIRI_VLR_OK]

	2. Valore out of Range [error]
--	--------------------------------

Parametro: Curve

Parametro	Curve
Formato	(curve != null)&&(curve.isEmpty())
Categorie	Formato
Formato [CURVE_FRM]	1. Non rispetta il formato [error] 2. Rispetta il formato [CURVE_FRM_OK]

Parametro: RETTILINEI

Parametro	Rettilinei\
Formato	(rettilinei != null)&&(rettilinei.isEmpty())
Categorie	Formato
Formato [RETT_FRM]	1. Non rispetta il formato [error] 2. Rispetta il formato [RETT_FRM_OK]

Test Case Name: TestGetNome
Precondizioni:
circuito.nome == “Red Bull Ring ”
Flusso di Eventi:
1. Alla variabile exp viene assegnata la stringa “Red Bull Ring ” 2. Alla variabile act viene assegnato il valore di ritorno circuito.getNome() 3. Chiamo il metodo assertEquals passando come parametri le variabili exp e act 1.

Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetName
Precondizioni:
circuito.nome == “Red Bull Ring ”
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Viene chiamato il metodo circuito.setNome(String) passando come parametro la stringa “Hungaroring” 2. Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo circuito.getNome() e la stringa “Hungaroring” <p>1.</p>
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestGetSede
Precondizioni:
circuito.sede == “Austria”
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Alla variabile exp viene assegnata la stringa “Austria” 2. Alla variabile act viene assegnato il valore di ritorno circuito.getSede() 3. Chiamo il metodo assertEquals passando come parametri le variabili exp e act <p>1.</p>

Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetSede
Precondizioni:
circuito.nome == "Austria "
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo circuito.setSede(String) passando come parametro la stringa "Ungheria" Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo circuito.getSede() e la stringa "Ungheria"
1.
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSedeNull
Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo circuito.setSede(String) passando come parametro il valore null Chiamo il metodo fail0;

1.
Oracolo:
non viene cenerata alcuna eccezione

Test Case Name: TestSedeEmpty
Precondizioni:
Flusso di Eventi:
<div> <div>5. Viene chiamato il metodo circuito.setSede(String) passando come parametro il valore ""</div> <div>6. Chiamo il metodo fail0;</div> <div>1.</div> </div>
Oracolo:
non viene generata alcuna eccezione

Test Case Name: TestGetLunghezza
Precondizioni:
circuito.lunghezza == 4318
Flusso di Eventi:
<div> <div>1. Alla variabile exp viene assegnato il valore 4318</div> <div>2. Alla variabile act viene assegnato il valore di ritorno circuito.getLunghezza()</div> <div>3. Chiamo il metodo assertEquals passando come parametri le variabili exp e act</div> </div>

Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetLunghezza
Precondizioni:
rettilineo.lunghezza == 4318
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo circuito.setLunghezza(int) passando come parametro l'intero 4500 Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo circuito.getLunghezza() e l'intero 4500
1.
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestLunghezzaRange
Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo rettilineo.setLunghezza (double) passando come parametro il valore -700.0 Chiamo il metodo fail0;

1.
Oracolo:
non viene cenerata alcuna eccezione

Test Case Name: TestGetMeteo
Precondizioni:
circuito.meteo == “sole”
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Alla variabile exp viene assegnata la stringa “sole” 2. Alla variabile act viene assegnato il valore di ritorno circuito.getMeteo() 3. Chiamo il metodo assertEquals passando come parametri le variabili exp e act
1.
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetMeteo
Precondizioni:
circuito.meteo == “sole”
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Viene chiamato il metodo circuito.setMeteo(String) passando come parametro la stringa “pioggia” 2. Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo circuito.getMeteo() e la stringa “pioggia”

1.
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestMeteoNull
Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo circuito.setMeteo(String) passando come parametro il valore null Chiamo il metodo fail0;
Oracolo:
non viene generata alcuna eccezione

Test Case Name: TestMeteoEmpty
Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo circuito.setMeteo(String) passando come parametro il valore "" Chiamo il metodo fail0;
Oracolo:
non viene generata alcuna eccezione

Test Case Name: TestGetTPM
Precondizioni:
circuito.TPM== 30
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Alla variabile exp viene assegnato il valore 30 2. Alla variabile act viene assegnato il valore di ritorno circuito.getTPM() 3. Chiamo il metodo assertEquals passando come parametri le variabili exp e act
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetTPM
Precondizioni:
rettilineo.TPM == 30
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Viene chiamato il metodo circuito.setTPM(int) passando come parametro l'intero 50 2. Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo circuito.getTPM() e l'intero 50
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestTPMRange
Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo rettilineo.setTPM (int) passando come parametro il valore -40 Chiamo il metodo fail0;
Oracolo:
non viene generata alcuna eccezione

Test Case Name: TestGetRPM
Precondizioni:
circuito.RPM == 50
Flusso di Eventi:
<ol style="list-style-type: none"> Alla variabile exp viene assegnato il valore 50 Alla variabile act viene assegnato il valore di ritorno circuito.getRPM() Chiamo il metodo assertEquals passando come parametri le variabili exp e act
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetTPM
Precondizioni:
rettilineo.lunghezza == 50
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo circuito.setRPM(int) passando come parametro l'intero 60 Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo circuito.get.RPM() e l'intero 60
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestTPMRange
Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo rettilineo.setRPM (int) passando come parametro il valore -40 Chiamo il metodo fail0;
Oracolo:
non viene generata alcuna eccezione

Test Case Name: TestGetUmidita

Precondizioni:
circuito.umidita == 12
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Alla variabile exp viene assegnato il valore 12 2. Alla variabile act viene assegnato il valore di ritorno circuito.getUmidita () 3. Chiamo il metodo assertEquals passando come parametri le variabili exp e act
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetUmidita
Precondizioni:
rettilineo.umidita == 12
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Viene chiamato il metodo circuito.setUmidita(int) passando come parametro l'intero 23 2. Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo circuito.getUmidita() e l'intero 23
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestUmiditaRange

Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo rettilineo.setUmidita (int) passando come parametro il valore -12 Chiamo il metodo fail0;
Oracolo:
non viene generata alcuna eccezione

Test Case Name: TestGetNumeroGiri
Precondizioni:
circuito.numeroGiri== 71
Flusso di Eventi:
<ol style="list-style-type: none"> Alla variabile exp viene assegnato il valore 71 Alla variabile act viene assegnato il valore di ritorno circuito.getNumeroGiri() Chiamo il metodo assertEquals passando come parametri le variabili exp e act
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetNumeroGiri
--

Precondizioni:
rettilineo.lunghezza == 71
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo circuito.setNumeroGiri(int) passando come parametro l'intero 91 Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo circuito.getNumeroGiri() e l'intero 91
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestNumeroGiriRange
Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo rettilineo.setNumeroGiri (int) passando come parametro il valore -50 Chiamo il metodo fail0;
Oracolo:
non viene generata alcuna eccezione

Test Case Name: TestGetCurve
Precondizioni:
<pre>circuito.curve == ArrayList<Curva> curve= { id= 1, nome= "curva1", angolo=30.0f) id= 2, nome= "curva2", angolo=35.0f) id= 3 nome= "curve3", angolo=35.0f) }</pre>
Flusso di Eventi:
<ol style="list-style-type: none"> Viene istanziata un ArrayList<Curve> exp contenente le curve <ul style="list-style-type: none"> id= 1, nome= "curva1", angolo=30.0f) id= 2, nome= "curva2", angolo=35.0f) id= 3 nome= "curva3", angolo=35.0f) Alla variabile act viene assegnato il valore di ritorno circuito.getCurve() Chiamo il metodo assertEquals passando come parametri le variabili exp e act
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetCurve
Precondizioni:
<pre>circuito.curve == ArrayList<Curva> curve= { id= 1, nome= "curva1", angolo=30.0f) id= 2, nome= "curva2", angolo=35.0f) id= 3 nome= "curve3", angolo=35.0f) }</pre>
Flusso di Eventi:

<ol style="list-style-type: none"> Viene istanziata un ArrayList<Curva> exp contenente le curve <ul style="list-style-type: none"> id= 5, nome= “curva4”, angolo=45.0f) id= 6, nome= “curva5”, angolo=55.0f) id= 7 nome= “curva6”, angolo=75.0f) Viene chiamato il metodo circuito.setCurve(ArrayList<Curva>) passando come parametro exp Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo circuito.getCurve() e la variabile exp
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestCurveNull
Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo circuito.setCurve(ArrayList<Curva>) passando come parametro il valore null Chiamo il metodo fail0;
Oracolo:
non viene generata alcuna eccezione

Test Case Name: TestCurveEmpty

Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo <code>circuito.setCurve(ArrayList<Curva>)</code> passando come parametro <code>new ArrayList<Curva>()</code>; Chiamo il metodo <code>fail0</code>;
Oracolo:
non viene generata alcuna eccezione

Test Case Name: TestGetCurve
Precondizioni:
<pre>circuito.rettilinei == ArrayList<Rettilineoe> rettilinei= { id= 1, nome= "rettilineo1", lunghezza=100.0) id= 2, nome= "rettilineo2", lunghezza=500.0) id= 3 nome= "rettilineo3", lunghezza=700.0)</pre>
Flusso di Eventi:
<ol style="list-style-type: none"> Viene istanziata un <code>ArrayList<Rettilineo></code> exp contenente le curve <pre>id= 1, nome= "rettilineo1", lunghezza=100.0) id= 2, nome= "rettilineo2", lunghezza=500.0) id= 3 nome= "rettilineo3", lunghezza=700.0)</pre> Alla variabile act viene assegnato il valore di ritorno <code>circuito.getRettilinei()</code> Chiamo il metodo <code>assertEquals</code> passando come parametri le variabili exp e act
Oracolo:
<code>assertEquals(exp,act) == true</code>

Test Case Name: TestSetRettilinei
Precondizioni:
<pre> circuito.rettilinei == ArrayList<Rettilineoe> rettilinei= { id= 1, nome= "rettilineo1", lunghezza=100.0) id= 2, nome= "rettilineo2", lunghezza=500.0) id= 3 nome= "rettilineo3", lunghezza=700.0) </pre>
Flusso di Eventi:
<ol style="list-style-type: none"> Viene istanziata un ArrayList<Rettilineo> exp contenente i rettilinei <pre> id= 1, nome= "rettilineo2", lunghezza=230.0 id= 1, nome= "rettilineo3", lunghezza=340.0 id= 1, nome= "rettilineo4", lunghezza=450.0 </pre> Viene chiamato il metodo circuito.setRettilieni(ArrayList<Rettilineo>) passando come parametro exp Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo circuito.getRettilineo() e la variabile exp
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestRettilineoNull
Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo circuito.setRettilinei(ArrayList<Rettilineo>) passando come parametro il valore null Chiamo il metodo fail0;
Oracolo:
non viene generata alcuna eccezione

Test Case Name: TestRettilineiEmpty
Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo circuito.setRettilinei(ArrayList<Rettilineo>) passando come parametro new ArrayList<Rettilineo>(); Chiamo il metodo fail0;
Oracolo:
non viene generata alcuna eccezione

4.1 Pilota

Scelta delle Categorie:

Parametro: NumeroPole

Parametro	Punteggio
Formato	(0<=punteggio<=575)
Categorie	Valore
VALORE[PUNTEGGIO_VLR]	<ul style="list-style-type: none">○ Valore in-Range [PUNTEGGIO_VLR_OK]○ Valore out of Range [error]

Parametro: NumeroVittorie

Parametro	NumeroVittorie
Formato	0<=numeroVittorie<=24
Categorie	Valore
VALORE[NUMEROVITTORIE_VLR]	<ul style="list-style-type: none">○ Valore in-Range [NUMEROVITTORIE_VLR_OK]○ Valore out of Range [error]

Parametro: NumeroPiazzamenti

Parametro	NumeroPiazzamenti
Formato	0<=numeroPiazzamenti<=24
Categorie	Valore
VALORE[NUMEROPIAZZAMENTI_VLR]	<ul style="list-style-type: none">○ Valore in-Range [NUMEROPIAZZAMENTI_VLR_OK]○ Valore out of Range [error]

Parametro: NumeroRitiri

Parametro	NumeroRitiri
Formato	0<=numeroRitiri<=24
Categorie	Valore
VALORE[NUMERORITIRI_VLR]	<ul style="list-style-type: none">○ Valore in-Range [NUMERORITIRI_VLR_OK]○ Valore out of Range [error]

Test Case Name: TestGetPunteggio

Precondizioni:
pilota.punteggio == 498
Flusso di Eventi:
<ul style="list-style-type: none"> 1. Alla variabile exp viene assegnato il valore 498 2. Alla variabile act viene assegnato il valore di ritorno pilota.getPunteggio() 3. Chiamo il metodo assertEquals passando come parametri le variabili exp e act
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetPunteggio
Precondizioni:
pilota.punteggio == 498
Flusso di Eventi:
1.1 Viene chiamato il metodo pilota.setPunteggio(500) 1.2 Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo pilota.getPunteggio() e 500
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestPunteggioRange
Precondizioni:
Flusso di Eventi:
1.1 Viene chiamato il metodo pilota.setPunteggio(-5000) 1.2 Chiamo il metodo fail();
Oracolo:
non viene generata alcuna eccezione

Test Case Name: TestGetNumeroVittorie
--

Precondizioni:
<code>pilota.numeroVittorie == 6</code>
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Alla variabile exp viene assegnato il valore 6 2. Alla variabile act viene assegnato il valore di ritorno <code>pilota.getNumeroVittorie()</code> 3. Chiamo il metodo <code>assertEquals</code> passando come parametri le variabili exp e act
Oracolo:
<code>assertEquals(exp,act) == true</code>

Test Case Name: TestSetNumeroVittorie
Precondizioni:
<code>pilota.numeroVittorie == 6</code>
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Viene chiamato il metodo <code>pilota.setNumeroVittorie(7)</code> 2. Chiamo il metodo <code>assertEquals</code> passando come parametri il valore di ritorno del metodo <code>pilota.getNumeroVittorie()</code> e 7
Oracolo:
<code>assertEquals(exp,act) == true</code>

Test Case Name: TestPunteggioRange
Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Viene chiamato il metodo <code>pilota.setNumeroVittorie(-5000)</code> 2. Chiamo il metodo <code>fail0;</code>
Oracolo:
non viene cenerata alcuna eccezione

Test Case Name: TestGetNumeroPole
Precondizioni:
<code>pilota.numeroPole == 7</code>

Flusso di Eventi:
<ol style="list-style-type: none"> 1. Alla variabile exp viene assegnato il valore 7 2. Alla variabile act viene assegnato il valore di ritorno pilota.getNumeroPole() 3. Chiamo il metodo assertEquals passando come parametri le variabili exp e act
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetNumeroPole
Precondizioni:
pilota.numeroPole == 7
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Viene chiamato il metodo pilota.setNumeroPole(7) 2. Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo pilota.getNumeroPole() e 7
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestNumeroPoleRange
Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Viene chiamato il metodo pilota.setNumeroPole(-5000) 2. Chiamo il metodo fail0;
Oracolo:
non viene cenerata alcuna eccezione

Test Case Name: TestGetNumeroPiazzamanti
Precondizioni:
pilota.numeroPiazzamenti == 14
Flusso di Eventi:

<ol style="list-style-type: none"> 1. Alla variabile exp viene assegnato il valore 14 2. Alla variabile act viene assegnato il valore di ritorno pilota.getNumeroPiazzamanti() 3. Chiamo il metodo assertEquals passando come parametri le variabili exp e act
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetNumeroPiazzamenti
Precondizioni:
pilota.numeroPiazzamanti == 14
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Viene chiamato il metodo pilota.setNumeroPiazzamanti(15) 2. Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo pilota.getNumeroPiazzamenti() e 15
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestNumeroPiazzamantiRange
Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Viene chiamato il metodo pilota.setNumeroPiazzamenti(-5000) 2. Chiamo il metodo fail();
Oracolo:
non viene generata alcuna eccezione

Test Case Name: TestGetNumeroRitiri
Precondizioni:
pilota.numeroRitiri == 0
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Alla variabile exp viene assegnato il valore 0 2. Alla variabile act viene assegnato il valore di ritorno pilota.getNumeroRitiri()

3. Chiamo il metodo assertEquals passando come parametri le variabili exp e act
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetNumeroRitiri
Precondizioni:
pilota.punteggio == 0
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo pilota.setNumeroRitiri(1) Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo pilota.getNumeroRitiri() e 1
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestNumeroRitiriRange
Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo pilota.setNumeroRitiri(-5000) Chiamo il metodo fail0;
Oracolo:
non viene generata alcuna eccezione

5.1 Setup

Scelta delle Categorie:

Parametro: CaricoAreodinamicoAnteriore

Parametro	CaricoAreodinamicoAnteriore
Formato	(0<=caricoAreodinamicoAnteriore<=10)
Categorie	Valore
VALORE[CARICOAREODINAMICOANTERI	10.1 Valore in-Range

ORE_VLR]	[CARICOAREODINAMICOANTERIORE_VLR_OK] 10.2 Valore out of Range [error]
----------	--

Parametro: CaricoAreodinamicoPosteriore

Parametro	CaricoAreodinamicoPosteriore
Formato	0<=caricoAreodinamicoPosteriore<=10
Categorie	Valore
VALORE[CARICOAREODINAMICOPOSTERIORE_VLR]	10.3 Valore in-Range [CARICOAREODINAMICOPOSTERIORE_VLR_OK] 10.4 Valore out of Range [error]

Parametro: CampanaturaAnteriore

Parametro	CampanaturaAnteriore
Formato	-5<=campanaturaAnteriore<=5
Categorie	Valore
VALORE[CAMPANATURAANTERIORE_VLR]	10.5 Valore in-Range [CAMPANATURAANTERIORE_VLR_OK] 10.6 Valore out of Range [error]

Parametro: CampanaturaPosteriore

Parametro	CampanaturaPosteriore
Formato	-5<=campanaturaPosteriore<=5
Categorie	Valore
VALORE[CAMPANATURAPOSTERIORE_VLR]	10.7 Valore in-Range [CAMPANATURAPOSTERIORE_VLR_OK] 10.8 Valore out of Range [error]

Parametro: PressioneFreni

Parametro	PressioneFreni
Formato	(0<=pressioneFreni<=100)
Categorie	Valore
VALORE[PRESSIONEFRENI_VLR]	2.1 Valore in-Range [PRESSIONEFRENI_VLR_OK] 2.2 Valore out of Range [error]

Parametro: BarraAntirollioAnteriore

Parametro	BarraAntirollioAnteriore
Formato	0<=barraAntirollioPosteriore<=10
Categorie	Valore
VALORE[BARRAANTIROLLIOPOSTERIORE_VLR]	2.3 Valore in-Range [BARRAANTIROLLIOPOSTERIORE_VLR_OK] 2.4 Valore out of Range [error]

Parametro: BarraAntirollioPosteriore

Parametro	BarraAntirollioPosteriore
Formato	-5<=campanaturaAnteriore<=5
Categorie	Valore
VALORE[CAMPANATURAANTERIORE_VLR]	2.5 Valore in-Range [CAMPANATURAANTERIORE_VLR_OK] 2.6 Valore out of Range [error]

Parametro: CampanaturaPosteriore

Parametro	CampanaturaPosteriore
Formato	-5<=campanaturaPosteriore<=5
Categorie	Valore
VALORE[CAMPANATURAPOSTERIORE_VLR]	2.7 Valore in-Range [CAMPANATURAPOSTERIORE_VLR_OK] 2.8 Valore out of Range [error]

Test Case Name: TestGetDate
Precondizioni:
setup.date == "10-10-2021"
Flusso di Eventi:
<ol style="list-style-type: none">1. Alla variabile exp viene assegnato il valore "10-10-2021"2. Alla variabile act viene assegnato il valore di ritorno setup.getDate()3. Chiamo il metodo assertEquals passando come parametri le variabili exp e act
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetDate
Precondizioni:
setup.date == "10-10-2021"
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Viene chiamato il metodo setup.setDate("11-11-2021") 2. Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo setup.getDate() e "11-11-2021"
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestGetId
Precondizioni:
setup.Id == 1
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Alla variabile exp viene assegnato il valore 1 2. Alla variabile act viene assegnato il valore di ritorno setup.getId() 3. Chiamo il metodo assertEquals passando come parametri le variabili exp e act
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetPunteggio
Precondizioni:
setup.id == 1
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Viene chiamato il metodo setup.setId(2) 2. Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo setup.getId() e 2
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestGetCaricoAreodinamicoAnteriore
Precondizioni:
setup.caricoAreodinamicoAnteriore == 5.3
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Alla variabile exp viene assegnato il valore 5.3 2. Alla variabile act viene assegnato il valore di ritorno setup.getCaricoAreodinamicoAnteriore() 3. Chiamo il metodo assertEquals passando come parametri le variabili exp e act
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetcaricoAreodinamicoAnteriore
Precondizioni:
setup.caricoAreodinamicoAnteriore == 5.3
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Viene chiamato il metodo setup.setCaricoAreodinamicoAnteriore(2) 2. Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo setup.getCaricoAreodinamicoAnteriore() e 2
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestPunteggioRange
Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Viene chiamato il metodo setup.setCaricoAreodinamicoAnteriore(-5000) 2. Chiamo il metodo fail0;
Oracolo:
non viene generata alcuna eccezione

Test Case Name: TestGetCaricoAreodinamicoPosteriore
Precondizioni:
setup.caricoAreodinamicoPosteriore == 5.2
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Alla variabile exp viene assegnato il valore 5.2 2. Alla variabile act viene assegnato il valore di ritorno setup.getCaricoAreodinamicoPosteriore() 3. Chiamo il metodo assertEquals passando come parametri le variabili exp e act
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetcaricoAreodinamicoPosteriore
Precondizioni:
setup.caricoAreodinamicoPosteriore == 5.2
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Viene chiamato il metodo setup.setCaricoAreodinamicoPosteriore(2) 2. Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo setup.getCaricoAreodinamicoPosteriore() e 2
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestPunteggioRange
Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Viene chiamato il metodo setup.setCaricoAreodinamicoPosteriore(-5000) 2. Chiamo il metodo fail0;
Oracolo:
non viene cenerata alcuna eccezione

Test Case Name: TestGetCampanaturaAnteriore
--

Precondizioni:
setup.campanaturaAnteriore == 1.5
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Alla variabile exp viene assegnato il valore 1.5 2. Alla variabile act viene assegnato il valore di ritorno setup.getCampanaturaAnteriore() 3. Chiamo il metodo assertEquals passando come parametri le variabili exp e act
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetCampanaturaAnteriore
Precondizioni:
setup.caricoAreodinamicoAnteriore == 1.5
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Viene chiamato il metodo setup.setCampanaturaAnteriore(1) 2. Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo setup.getCampanaturaAnteriore() e 1
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestCampanaturaAnterioreRange
Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Viene chiamato il metodo setup.setCampanaturaAnteriore(-5000) 2. Chiamo il metodo fail0;
Oracolo:
non viene generata alcuna eccezione

Test Case Name: TestGetCampanaturaPosteriore
Precondizioni:

setup.campanaturaPosteriore == 1.3
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Alla variabile exp viene assegnato il valore 1.3 2. Alla variabile act viene assegnato il valore di ritorno setup.getCampanaturaPosteriore() 3. Chiamo il metodo assertEquals passando come parametri le variabili exp e act
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetCampanaturaPosteriore
Precondizioni:
setup.campanaturaPosteriore == 1.3
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Viene chiamato il metodo setup.setCampanaturaPosteriore (1) 2. Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo setup.getCampanaturaPosteriore() e 1
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestPunteggioRange
Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Viene chiamato il metodo setup.setCampanaturaPosteriore(-5000) 2. Chiamo il metodo fail0;
Oracolo:
non viene cenerata alcuna eccezione

Test Case Name: TestGetConvergenzaAnteriore
Precondizioni:
setup.convergenzaAnteriore == 0.6

Flusso di Eventi:
<ol style="list-style-type: none"> 1. Alla variabile exp viene assegnato il valore 0.6 2. Alla variabile act viene assegnato il valore di ritorno setup.getConvergenzaAnteriore() 3. Chiamo il metodo assertEquals passando come parametri le variabili exp e act
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetConvergenzaAnteriore
Precondizioni:
setup.convergenzaAnteriore == 0.6
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Viene chiamato il metodo setup.set(0) 2. Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo setup.getConvergenzaAnteriore() e 0
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestConevergenzaAnterioreRange
Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Viene chiamato il metodo setup.setConvergenzaAnteriore(-5000) 2. Chiamo il metodo fail0;
Oracolo:
non viene generata alcuna eccezione

Test Case Name: TestGetConvergenzaPosteriore
Precondizioni:
setup.convergenzaPosteriore == 0.3
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Alla variabile exp viene assegnato il valore 0.3

2. Alla variabile act viene assegnato il valore di ritorno setup.getConvergenzaPosteriore()
3. Chiamo il metodo assertEquals passando come parametri le variabili exp e act

Oracolo:

assertEquals(exp,act) == true

Test Case Name: TestSetConvergenzaPosteriore

Precondizioni:

setup.convergenzaPosteriore ==0.3

Flusso di Eventi:

1. Viene chiamato il metodo setup.setConvergenzaPosteriore(0)
2. Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo setup.getConvergenzaPosteriore() e 0

Oracolo:

assertEquals(exp,act) == true

Test Case Name: TestConvergenzaPosterioreRange

Precondizioni:

Flusso di Eventi:

1. Viene chiamato il metodo setup.setConvergenzaPosteriore(-5000)
2. Chiamo il metodo fail();

Oracolo:

non viene cenerata alcuna eccezione

Test Case Name: TestGetPressioneFreni

Precondizioni:

setup.pressioneFreni == 50

Flusso di Eventi:

1. Alla variabile exp viene assegnato il valore 50
2. Alla variabile act viene assegnato il valore di ritorno setup.getPressioneFreni()

3. Chiamo il metodo assertEquals passando come parametri le variabili exp e act
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetPressioneFreni
Precondizioni:
setup.pressioneFreni == 50
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo setup.setPressioneFreni(15) Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo setup.getPressioneFreni() e 15
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestPressioneFreni
Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo setup.setPressioneFreni(-5000) Chiamo il metodo fail0;
Oracolo:
non viene generata alcuna eccezione

Test Case Name: TestGetBarraAntirollioAnteriore
Precondizioni:
setup.barraAntirollioAnteriore == 5.8
Flusso di Eventi:
<ol style="list-style-type: none"> Alla variabile exp viene assegnato il valore 5.8 Alla variabile act viene assegnato il valore di ritorno setup.getBarraAntirollioAnteriore()

3. Chiamo il metodo assertEquals passando come parametri le variabili exp e act
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetBarraAntirollioAnteriore
Precondizioni:
setup.barraAntirollioAnteriore == 5.3
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo setup.setBarraAntirollioAnteriore(5) Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo setup.getBarraAntirollioAnteriore() e 5
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestBarraAntirollioAnterioreRange
Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo setup.setBarraAntirollioAnteriore(-5000) Chiamo il metodo fail0;
Oracolo:
non viene generata alcuna eccezione

Test Case Name: TestGetBarraAntirollioPosteriore
Precondizioni:
setup.barraAntirollioPosterire == 5.5
Flusso di Eventi:
<ol style="list-style-type: none"> Alla variabile exp viene assegnato il valore 5.5 Alla variabile act viene assegnato il valore di ritorno setup.getBarraAntirollioPosteriore() Chiamo il metodo assertEquals passando come parametri le variabili exp e act

Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetBarraAntirollioPosteriore
Precondizioni:
setup.barraAntirollioPosteriore == 5.3
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo setup.setCaricoAreodinamicoAnteriore(5.5) Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo setup.getBarraAntirollioPosteriore() e 5.5
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestBarraAntirollioPosterioreRange
Precondizioni:
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo setup.setBarraAntirollioPosteriore(-5000) Chiamo il metodo fail0;
Oracolo:
non viene generata alcuna eccezione

6.1 Tecnico

Scelta delle Categorie:

Parametro: SetupList

Parametro	SetupList
Formato	setupList != null
Categorie	Formato

Formato [SETUPLIST_FRM]	<ul style="list-style-type: none"> • Non rispetta il formato [error] • Rispetta il formato [SETUPLIST_FRM_OK]
-------------------------	---

Parametro: Setup

Parametro	Setup
Formato	setup != null
Categorie	Formato
Formato [SETUP_FRM]	<ol style="list-style-type: none"> 1 Non rispetta il formato [error] 2 Rispetta il formato [SETUP_FRM_OK]

Test Case Name: TestGetSetupList
Precondizioni:
tecnico.setupList == [...]
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Alla variabile exp viene assegnato il riferimento a setupList 2. Alla variabile act viene assegnato il valore di ritorno tecnico.getSetupList() 3. Chiamo il metodo assertEquals passando come parametri le variabili exp e act
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetSetupList
Precondizioni:
tecnico.setupList == [...]
Flusso di Eventi:
<ol style="list-style-type: none"> 1. Vengono aggiunti due setup a setupList2 e successivamente viene chiamato tecnico.setSetupList(setupList2) 2. Chiamo il metodo assertEquals passando come parametri il valore di ritorno del metodo tecnico.getSetupList() e setupList2 3. Ripristiniamo con tecnico.setSetupList(setupList)
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestSetupListNull

Precondizioni:
tecnico.setupList == [...]
Flusso di Eventi:
<ol style="list-style-type: none"> Viene chiamato il metodo tecnico.setSetupList (null) Chiamo il metodo fail0; Ripristino con tecnico.setSetupList(setupList)
Oracolo:
non viene generata alcuna eccezione

Test Case Name: TestAddSetup
Precondizioni:
tecnico.setupList == [...]
Flusso di Eventi:
<ol style="list-style-type: none"> 2.1 Creo un nuovo setup che chiamo setupToAdd 2.2 Chiamo tecnico.addSetup(setupToAdd) 2.3 Chiamo il metodo assertEquals passando come parametri true e tecnico.getSetupList().contains(setupToAdd)
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestRemoveSetup
Precondizioni:
tecnico.setupList == [...]
Flusso di Eventi:
<ol style="list-style-type: none"> 2.1 Creo un nuovo setup che chiamo setupToRemove 2.2 Chiamo tecnico.removeSetup(setupToAdd) 2.3 Chiamo il metodo assertEquals passando come parametri 0 e tecnico.getSetupList().contains(setupToRemove)
Oracolo:
assertEquals(exp,act) == true

Test Case Name: TestAddSetupNull

Precondizioni:
tecnico.setupList == [...]
Flusso di Eventi:
2.1 Chiamo tecnico.addSetup(null) 2.2 Chiamo il metodo fail0;
Oracolo:
non viene generata alcuna eccezione

Test Case Name: TestRemoveSetupNull
Precondizioni:
tecnico.setupList == [...]
Flusso di Eventi:
2.1 Chiamo tecnico.removeSetup(null) 2.2 Chiamo il metodo fail0;
Oracolo:
non viene generata alcuna eccezione