

SiVoDiM

Sintesi Vocale per Dispositivi Mobili



Definizione di prodotto

Versione	2.0.0
Redattori	Francesco Bizzaro Gino Zaidan Riccardo Rizzo
Verificatori	Alberto Andriolo Federico Rossetto
Responsabili	Gino Zaidan
Uso	Esterno
Lista di distribuzione	Stark Labs Prof. Tullio Vardanega, Prof. Riccardo Cardin

Architettura di dettaglio relativa al progetto SiVoDiM

Registro delle modifiche

Versione	Data	Attività	Autori
2.0.0	06/07/2016	Accettazione	Gino Zaidan
1.2.0	06/07/2016	Revisione della sezione dedicata al package Libraries	Alberto Andriolo
1.1.6	06/07/2016	Giustificato uso di metodi protected nella classe MivoqTTSService	Riccardo Rizzo
1.1.5	06/07/2016	Giustificato uso di attributo statico nella classe Enginelmpl	Riccardo Rizzo
1.1.4	06/07/2016	Migliorati i contenuti descrittivi delle classi e dei metodi del package Libraries	Riccardo Rizzo
1.1.3	06/07/2016	Giustificata la scelta implementativa dei Presenter nella descrizione dei Package	Riccardo Rizzo
1.1.2	06/07/2016	Corretti gli attributi statici nel diagramma di MivoqConnectionImpl	Riccardo Rizzo
1.1.1	06/07/2016	Corretti i metodi nei diagrammi di MivoqConnectionFactory	Riccardo Rizzo
1.1.0	05/07/2016	Revisione delle modifiche	Federico Rossetto
1.0.6	05/07/2016	Aggiunti nuovi diagrammi di sequenza	Francesco Bizzaro
1.0.5	05/07/2016	Riordinamento delle classi per garantire una migliore fruizione del documento	Riccardo Rizzo
1.0.4	05/07/2016	Aggiunte nuove classi nel package Drama::Presenter	Riccardo Rizzo
1.0.3	05/07/2016	Aggiunte nuove classi nel package Drama::Model::Utilities	Gino Zaidan
1.0.2	05/07/2016	Aggiunte nuove classi nel package Drama::Model::Screenplay	Riccardo Rizzo
1.0.1	04/07/2016	Rimozione classi di librerie esterne	Gino Zaidan
1.0.0	31/05/2016	Accettazione	Federico Rossetto
0.4.0	31/05/2016	Revisione finale	Federico Rossetto
0.3.2	30/05/2016	Stesura Appendice Tracciamento	Enrico Chiara

0.3.1	30/05/2016	Stesura sezione 4 Diagrammi di sequenza	Francesco Bizzaro
0.3.0	30/05/2016	Verifica della sezione 3 Specifica componenti	Alberto Andriolo
0.2.1	27/05/2016	Stesura delle parti che descrivono la componente Sivodim::Drama nella sezione 3 Specifica componenti	Francesco Bizzaro, Riccardo Rizzo
0.2.0	26/05/2016	Verifica	Bizzaro Francesco
0.1.1	26/05/2016	Correzione degli errori segnalati	Alberto Andriolo
0.1.0	26/05/2016	Verifica generale del documento	Francesco Bizzaro
0.0.5	26/05/2016	Stesura delle parti che descrivono la componente Sivodim::Libraries nella sezione 3 Specifica componenti	Alberto Andriolo, Federico Rossetto
0.0.4	25/05/2016	Realizzazione dei diagrammi delle classi	Enrico Chiara
0.0.3	25/05/2016	Realizzazione dei diagrammi delle classi	Gino Zaidan
0.0.2	24/05/2016	Stesura sezioni 1 Introduzione e 2 Standard di progetto	Riccardo Rizzo
0.0.1	24/05/2016	Stesura struttura documento	Riccardo Rizzo

Indice

1 Introduzione	1
1.1 Scopo del documento	1
1.2 Scopo del progetto	1
1.3 Glossario	1
1.4 Riferimenti	1
1.4.1 Normativi	1
1.4.2 Informativi	2
2 Standard di progetto	3
2.1 Standard per la progettazione	3
2.2 Standard per la documentazione del codice	3
2.3 Standard di denominazione di entità e relazioni	3
2.4 Standard di programmazione	3
2.5 Strumenti di lavoro	3
3 Specifica componenti	4
3.1 Sivodim	4
3.1.1 Informazioni sul package	4
3.2 Sivodim::Libraries	4
3.2.1 Informazioni sul package	4
3.3 Sivodim::Libraries::Model	5
3.3.1 Informazioni sul package	5
3.4 Sivodim::Libraries::Model::EngineManager	5
3.4.1 Informazioni sul package	5
3.4.2 Engine (interface)	6
3.4.2.1 Descrizione	6
3.4.2.2 Utilizzo	6
3.4.2.3 Classi figlie	6
3.4.2.4 Metodi	6
3.4.2.4.1 createVoice(String name, String gender, String myLanguage)	7
3.4.2.4.2 getVoices()	7
3.4.2.4.3 removeVoice(int index)	7
3.4.2.4.4 speak(String voiceID, String text)	7
3.4.2.4.5 synthesizeToFile(String path, String voiceID, String myEmotion, String text, Engine.listener myListener)	8
3.4.2.4.6 getIsConnected()	8
3.4.2.4.7 save()	8
3.4.2.4.8 load()	8
3.4.2.4.9 getVoiceByName(String s)	9
3.4.2.4.10 setDefaultVoice(int pos)	9
3.4.3 Engine.Listener (interface)	9
3.4.3.1 Descrizione	9
3.4.3.2 Utilizzo	9
3.4.3.3 Metodi	9
3.4.3.3.1 onCompleteSynthesis()	9

3.4.4	Enginelmpl (class)	10
3.4.4.1	Descrizione	10
3.4.4.2	Utilizzo	10
3.4.4.3	Classi ereditate	10
3.4.4.4	Attributi	10
3.4.4.4.1	isConnected	10
3.4.4.4.2	myContext	10
3.4.4.4.3	backupEngine	11
3.4.4.4.4	myEngine	11
3.4.4.5	Metodi	11
3.4.4.5.1	Enginelmpl(Context c)	11
3.4.4.5.2	createVoice(String name, String gender, String myLanguage)	11
3.4.4.5.3	getVoices()	12
3.4.4.5.4	removeVoice(int index)	12
3.4.4.5.5	speak(String voiceID, String text)	12
3.4.4.5.6	synthesizeToFile(String path, String voiceID, String myEmotion, String text, Engine.listener myListener)	12
3.4.4.5.7	getIsConnected()	13
3.4.4.5.8	save()	13
3.4.4.5.9	load()	13
3.4.4.5.10	getVoiceByName(String s)	13
3.4.4.5.11	setDefaultVoice(int pos)	13
3.5	Sivodim::Libraries::Model::Integration	14
3.5.1	Informazioni sul package	14
3.5.2	CheckVoiceData (class)	14
3.5.2.1	Descrizione	14
3.5.2.2	Utilizzo	14
3.5.2.3	Metodi	14
3.5.2.3.1	CheckVoiceData()	15
3.5.2.3.2	getVoices()	15
3.5.2.3.3	onCreate(Bundle savedInstanceState)	15
3.5.3	DownloadVoiceData (class)	15
3.5.3.1	Descrizione	15
3.5.3.2	Utilizzo	15
3.5.3.3	Attributi	15
3.5.3.3.1	MTts	16
3.5.3.4	Metodi	16
3.5.3.4.1	DownloadVoiceData()	16
3.5.3.4.2	onInit(int status)	16
3.5.3.4.3	onCreate(Bundle savedInstanceState)	16
3.5.4	GetSampleText (class)	16
3.5.4.1	Descrizione	16
3.5.4.2	Utilizzo	17
3.5.4.3	Metodi	17
3.5.4.3.1	GetSampleText()	17
3.5.4.3.2	onCreate(Bundle savedInstanceState)	17
3.6	Sivodim::Libraries::Model::Mivoq	17

3.6.1	Informazioni sul package	17
3.6.2	AbstractFactory (abstract)	18
3.6.2.1	Descrizione	18
3.6.2.2	Utilizzo	18
3.6.2.3	Classi figlie	18
3.6.2.4	Metodi	19
3.6.2.4.1	createConnection()	19
3.6.3	MivoqConnectionFactory (class)	19
3.6.3.1	Descrizione	19
3.6.3.2	Utilizzo	19
3.6.3.3	Classi ereditate	19
3.6.3.4	Metodi	19
3.6.3.4.1	MivoqConnectionFactory()	19
3.6.3.4.2	createConnection()	19
3.6.4	MivoqConnection (interface)	20
3.6.4.1	Descrizione	20
3.6.4.2	Utilizzo	20
3.6.4.3	Classi figlie	20
3.6.4.4	Metodi	20
3.6.4.4.1	getResponse()	20
3.6.4.4.2	sendRequest(String text)	20
3.6.4.4.3	setEffects(String s)	21
3.6.4.4.4	setLocale(String s)	21
3.6.4.4.5	setQueue(RequestQueue rq)	21
3.6.4.4.6	setVoiceGender(String s)	21
3.6.4.4.7	setVoiceName(String s)	22
3.6.5	MivoqConnectionImpl (class)	22
3.6.5.1	Descrizione	22
3.6.5.2	Utilizzo	22
3.6.5.3	Classi ereditate	22
3.6.5.4	Attributi	22
3.6.5.4.1	effects	23
3.6.5.4.2	inputType	23
3.6.5.4.3	locale	23
3.6.5.4.4	myContext	23
3.6.5.4.5	myRequestQueue	23
3.6.5.4.6	outputFormat	23
3.6.5.4.7	outputType	24
3.6.5.4.8	request	24
3.6.5.4.9	response	24
3.6.5.4.10	selectAlgorithm	24
3.6.5.4.11	url	24
3.6.5.4.12	voiceGener	24
3.6.5.4.13	voiceName	25
3.6.5.5	Metodi	25
3.6.5.5.1	getResponse()	25
3.6.5.5.2	sendRequest(String text)	25
3.6.5.5.3	setEffects(String s)	25
3.6.5.5.4	setLocale(String s)	26

3.6.5.5.5	setQueue(RequestQueue rq)	26
3.6.5.5.6	setVoiceGender(String s)	26
3.6.5.5.7	setVoiceName(String s)	26
3.6.5.5.8	MivoqConnectionImpl()	27
3.6.6	MivoqTTSSingleton (class)	27
3.6.6.1	Descrizione	27
3.6.6.2	Utilizzo	27
3.6.6.3	Attributi	27
3.6.6.3.1	myContext	27
3.6.6.3.2	myFactory	27
3.6.6.3.3	ourInstance	28
3.6.6.3.4	queue	28
3.6.6.3.5	voiceList	28
3.6.6.4	Metodi	28
3.6.6.4.1	MivoqTTSSingleton()	28
3.6.6.4.2	createVoice(String name, String gender, String myLanguage)	28
3.6.6.4.3	getInstance()	29
3.6.6.4.4	getVoices()	29
3.6.6.4.5	hasContext()	29
3.6.6.4.6	setContext(Context T)	29
3.6.6.4.7	removeVoice(int index)	29
3.6.6.4.8	speak(MivoqVoice v, String text)	30
3.6.6.4.9	synthesizeToFile(String path, MivoqVoice v, String text)	30
3.6.6.4.10	synthesizeText(MivoqVoice v, String text)	30
3.6.7	MivoqTTSService (class)	30
3.6.7.1	Descrizione	31
3.6.7.2	Utilizzo	31
3.6.7.3	Classi ereditate	31
3.6.7.4	Attributi	31
3.6.7.4.1	mCallback	31
3.6.7.5	Metodi	31
3.6.7.5.1	MivoqTTSService()	31
3.6.7.5.2	onGetLanguage()	31
3.6.7.5.3	onIsLanguageAvailable(String lang, String country, String variant)	32
3.6.7.5.4	onLoadLanguage(String lang, String country, String variant)	32
3.6.7.5.5	onStop()	32
3.6.7.5.6	onSynthesizeText(SynthesisRequest req, SynthesisCallback call)	32
3.6.8	AudioRequest (class)	33
3.6.8.1	Descrizione	33
3.6.8.2	Utilizzo	33
3.6.8.3	Attributi	33
3.6.8.3.1	responseHeaders	33
3.6.8.3.2	mParams	33
3.6.8.3.3	delivered	33

3.6.8.3.4	mListener	34
3.6.8.3.5	mErrorListener	34
3.6.8.4	Metodi	34
3.6.8.4.1	AudioRequest(int post, String mUrl, Map<String, String> params, Response.Listener<byte[]> listener, Response.ErrorListener errorListener)	34
3.6.8.4.2	deliverResponse(byte[] response)	34
3.6.8.4.3	getHeaders()	35
3.6.8.4.4	getParams()	35
3.6.8.4.5	parseNetworkResponse(NetworkResponse response)	35
3.6.9	AuthAudioRequest (class)	35
3.6.9.1	Descrizione	35
3.6.9.2	Utilizzo	35
3.6.9.3	Attributi	36
3.6.9.3.1	username	36
3.6.9.3.2	password	36
3.6.9.3.3	responseHeaders	36
3.6.9.3.4	mParams	36
3.6.9.3.5	delivered	36
3.6.9.3.6	mListener	37
3.6.9.3.7	mErrorListener	37
3.6.9.4	Metodi	37
3.6.9.4.1	AuthAudioRequest(int post, String mUrl, Map<String, String> params, Response.Listener<byte[]> listener, Response.ErrorListener errorListener)	37
3.6.9.4.2	deliverResponse(byte[] response)	37
3.6.9.4.3	getHeaders()	38
3.6.9.4.4	getParams()	38
3.6.9.4.5	parseNetworkResponse(NetworkResponse response)	38
3.7	Sivodim::Libraries::Model::Voice	38
3.7.1	Informazioni sul package	38
3.7.2	MivoqVoice (class)	39
3.7.2.1	Descrizione	39
3.7.2.2	Utilizzo	39
3.7.2.3	Attributi	39
3.7.2.3.1	effects	39
3.7.2.3.2	femaleDeAddingEffects	39
3.7.2.3.3	gender	40
3.7.2.3.4	lang	40
3.7.2.3.5	myEmotion	40
3.7.2.3.6	name	40
3.7.2.3.7	voiceName	40
3.7.2.4	Metodi	40
3.7.2.4.1	MivoqVoice(String name, String myVoiceName, Language locale)	41
3.7.2.4.2	getEffects()	41
3.7.2.4.3	getEncodedName(String gen, String myLang)	41

3.7.2.4.4	getGender()	41
3.7.2.4.5	getLanguage()	41
3.7.2.4.6	getName()	42
3.7.2.4.7	getSampleText(String myLang)	42
3.7.2.4.8	getStringEffects()	42
3.7.2.4.9	getVoiceName()	42
3.7.2.4.10	removeEffect(int index)	42
3.7.2.4.11	setEffect(Effect e)	43
3.7.2.4.12	setEmotion(Emotion e)	43
3.7.2.4.13	setFemaleDe(boolean b)	43
3.7.2.4.14	setGender(String g)	43
3.7.2.4.15	setGenderLanguage(String gen, String myLang)	44
3.7.2.4.16	setName(String n)	44
3.7.3	Language (class)	44
3.7.3.1	Descrizione	44
3.7.3.2	Utilizzo	44
3.7.3.3	Attributi	44
3.7.3.3.1	lang	45
3.7.3.3.2	state	45
3.7.3.3.3	modifier	45
3.7.3.4	Metodi	45
3.7.3.4.1	Language(String l)	45
3.7.3.4.2	setLanguage(String l)	45
3.7.3.4.3	setState(String s)	46
3.7.3.4.4	setModifier(String m)	46
3.7.3.4.5	toString()	46
3.7.4	Emotion (class)	46
3.7.4.1	Descrizione	46
3.7.4.2	Utilizzo	47
3.7.4.3	Attributi	47
3.7.4.3.1	effects	47
3.7.4.3.2	Anger	47
3.7.4.3.3	Disgust	47
3.7.4.3.4	Fear	47
3.7.4.3.5	Happiness	47
3.7.4.3.6	Sadness	48
3.7.4.3.7	Surprise	48
3.7.4.4	Metodi	48
3.7.4.4.1	Emotion(String s)	48
3.7.4.4.2	toString()	48
3.7.5	Effect (interface)	48
3.7.5.1	Descrizione	49
3.7.5.2	Utilizzo	49
3.7.5.3	Classi figlie	49
3.7.5.4	Metodi	49
3.7.5.4.1	getName()	49
3.7.5.4.2	getValue()	49
3.7.5.4.3	setValue(String f)	49
3.7.5.4.4	toString()	49

3.7.6	EffectImpl (class)	50
3.7.6.1	Descrizione	50
3.7.6.2	Utilizzo	50
3.7.6.3	Classi ereditate	50
3.7.6.4	Attributi	50
3.7.6.4.1	effectName	50
3.7.6.4.2	value	50
3.7.6.5	Metodi	50
3.7.6.5.1	getName()	51
3.7.6.5.2	getValue()	51
3.7.6.5.3	setValue(String f)	51
3.7.6.5.4	toString()	51
3.7.6.5.5	EffectImpl(String name)	51
3.8	Sivodim::Libraries::Presenter	52
3.8.1	Informazioni sul package	52
3.8.2	HomePresenter (interface)	52
3.8.2.1	Descrizione	52
3.8.2.2	Utilizzo	52
3.8.2.3	Classi figlie	52
3.8.2.4	Metodi	53
3.8.2.4.1	goToVoiceListActivity(Context context)	53
3.8.2.4.2	goToNewVoiceActivity(Context context)	53
3.8.3	HomePresenterImpl (class)	53
3.8.3.1	Descrizione	53
3.8.3.2	Utilizzo	53
3.8.3.3	Classi ereditate	53
3.8.3.4	Attributi	54
3.8.3.4.1	engine	54
3.8.3.5	Metodi	54
3.8.3.5.1	HomePresenterImpl(Context context)	54
3.8.3.5.2	goToNewVoiceActivity(Context context)	54
3.8.3.5.3	goToVoiceListActivity(Context context)	54
3.8.4	VoicePresenter (interface)	55
3.8.4.1	Descrizione	55
3.8.4.2	Utilizzo	55
3.8.4.3	Classi figlie	55
3.8.4.4	Metodi	55
3.8.4.4.1	setActivity(NewVoiceActivityInterface newVoiceActivityInterface)	55
3.8.4.4.2	setActivity(EditVoiceActivityInterface newVoiceActivityInterface)	56
3.8.4.4.3	loadVoiceGender(Context context)	56
3.8.4.4.4	getGenderAdapter(Context context)	56
3.8.4.4.5	loadVoiceLanguage(Context context)	56
3.8.4.4.6	getLanguageAdapter(Context context)	57
3.8.4.4.7	getVoiceName()	57
3.8.5	VoicePresenterImpl (class)	57
3.8.5.1	Descrizione	57
3.8.5.2	Utilizzo	57

3.8.5.3	Classi ereditate	57
3.8.5.4	Attributi	58
3.8.5.4.1	editVoiceActivityInterface	58
3.8.5.4.2	engine	58
3.8.5.4.3	genderAdapter	58
3.8.5.4.4	languageAdapter	58
3.8.5.4.5	mivoqVoice	58
3.8.5.4.6	newVoiceActivityInterface	59
3.8.5.5	Metodi	59
3.8.5.5.1	VoicePresenterImpl(Engine engine)	59
3.8.5.5.2	VoicePresenterImpl(Mivoq mivoq)	59
3.8.5.5.3	setActivity(NewVoiceActivityInterface newVoiceActivityInterface)	59
3.8.5.5.4	setActivity(EditVoiceActivityInterface newVoiceActivityInterface)	60
3.8.5.5.5	loadVoiceGender(Context context)	60
3.8.5.5.6	getGenderAdapter(Context context)	60
3.8.5.5.7	loadVoiceLanguage(Context context)	60
3.8.5.5.8	getLanguageAdapter(Context context)	61
3.8.5.5.9	getVoiceName()	61
3.8.6	VoiceListPresenter (interface)	61
3.8.6.1	Descrizione	61
3.8.6.2	Utilizzo	61
3.8.6.3	Classi figlie	61
3.8.6.4	Metodi	61
3.8.6.4.1	goToEditVoiceActivity(Context context, MivoqVoice mivoqVoice)	62
3.8.6.4.2	setActivity(VoiceListActivityInterface voiceListActivityInterface)	62
3.8.6.4.3	createMivoqVoice(String s)	62
3.8.6.4.4	loadVoiceNames(Context context)	62
3.8.6.4.5	getVoicesAdapter(Context context)	63
3.8.6.4.6	goToEditVoiceActivity(Context context, MivoqVoice mivoqVoice)	63
3.8.7	VoiceListPresenterImpl (class)	63
3.8.7.1	Descrizione	63
3.8.7.2	Utilizzo	63
3.8.7.3	Classi ereditate	64
3.8.7.4	Attributi	64
3.8.7.4.1	engine	64
3.8.7.4.2	voiceListActivityInterface	64
3.8.7.4.3	voicesAdapterName	64
3.8.7.5	Metodi	64
3.8.7.5.1	VoiceListPresenterImpl(Engine engine)	64
3.8.7.5.2	setActivity(VoiceListActivityInterface voiceListActivityInterface)	65
3.8.7.5.3	createMivoqVoice(String s)	65
3.8.7.5.4	loadVoiceNames(Context context)	65
3.8.7.5.5	getVoicesAdapter(Context context)	65

3.9	Sivodim::Libraries::View	66
3.9.1	Informazioni sul package	66
3.9.2	HomeActivityInterface (interface)	66
3.9.2.1	Descrizione	66
3.9.2.2	Utilizzo	66
3.9.2.3	Classi figlie	66
3.9.3	HomeActivity (class)	67
3.9.3.1	Descrizione	67
3.9.3.2	Utilizzo	67
3.9.3.3	Classi ereditate	67
3.9.3.4	Attributi	67
3.9.3.4.1	homePresenter	67
3.9.4	NewVoiceActivityInterface (interface)	67
3.9.4.1	Descrizione	67
3.9.4.2	Utilizzo	68
3.9.4.3	Classi figlie	68
3.9.5	NewVoiceActivity (class)	68
3.9.5.1	Descrizione	68
3.9.5.2	Utilizzo	68
3.9.5.3	Classi ereditate	68
3.9.5.4	Attributi	68
3.9.5.4.1	genderAdapter	68
3.9.5.4.2	languageAdapter	69
3.9.5.4.3	voicePresenter	69
3.9.6	EditVoiceActivityInterface (interface)	69
3.9.6.1	Descrizione	69
3.9.6.2	Utilizzo	69
3.9.6.3	Classi figlie	69
3.9.7	EditVoiceActivity (class)	70
3.9.7.1	Descrizione	70
3.9.7.2	Utilizzo	70
3.9.7.3	Classi ereditate	70
3.9.7.4	Attributi	70
3.9.7.4.1	genderAdapter	70
3.9.7.4.2	languageAdapter	70
3.9.7.4.3	voicePresenter	71
3.9.8	VoiceListActivityInterface (interface)	71
3.9.8.1	Descrizione	71
3.9.8.2	Utilizzo	71
3.9.8.3	Classi figlie	71
3.9.9	VoiceListActivity (class)	71
3.9.9.1	Descrizione	71
3.9.9.2	Utilizzo	71
3.9.9.3	Classi ereditate	72
3.9.9.4	Attributi	72
3.9.9.4.1	voiceListAdapter	72
3.9.9.4.2	voiceListPresenter	72
3.9.9.4.3	voiceListView	72
3.10	Sivodim::Drama	72

3.10.1	Informazioni sul package	72
3.11	Sivodim::Drama::Model	73
3.11.1	Informazioni sul package	73
3.12	Sivodim::Drama::Presenter	74
3.12.1	Informazioni sul package	74
3.12.2	HomePresenter (interface)	75
3.12.2.1	Descrizione	75
3.12.2.2	Utilizzo	75
3.12.2.3	Classi figlie	75
3.12.2.4	Metodi	76
3.12.2.4.1	getTitlesAdapter(Context context)	76
3.12.2.4.2	getScreenplayTitles(Context context)	76
3.12.2.4.3	goToListChapter(Context context, String selected)	76
3.12.3	HomePresenterImpl (class)	77
3.12.3.1	Descrizione	77
3.12.3.2	Utilizzo	77
3.12.3.3	Classi ereditate	77
3.12.3.4	Attributi	77
3.12.3.4.1	titleAdapter	77
3.12.3.4.2	homeInterface	77
3.12.3.5	Metodi	77
3.12.3.5.1	HomePresenterImpl(HomeInterface homeInterface)	78
3.12.3.5.2	getScreenplayTitles(Context context)	78
3.12.3.5.3	getTitlesAdapter(Context context)	78
3.12.3.5.4	goToListChapter(Context context, String selected)	78
3.12.4	ScreenplayPresenter (interface)	79
3.12.4.1	Descrizione	79
3.12.4.2	Utilizzo	79
3.12.4.3	Classi figlie	79
3.12.4.4	Metodi	79
3.12.4.4.1	setActivity(ListChapterInterface listChapterInterface)	79
3.12.4.4.2	setActivity(NewChapterInterface newChapterInterface)	80
3.12.4.4.3	getTitlesAdapter(Context context, String screenplay)	80
3.12.4.4.4	getScreenplayTitle()	80
3.12.4.4.5	getScreenplay()	80
3.12.4.4.6	export()	81
3.12.4.4.7	share()	81
3.12.4.4.8	newChapter(String title)	81
3.12.4.4.9	newScreenplay(String title, Context context)	81
3.12.4.4.10	moveUpChapter(String chapterTitle)	81
3.12.4.4.11	moveDownChapter(String chapterTitle)	82
3.12.4.4.12	save(Screenplay screenplay, Context context)	82
3.12.4.4.13	addCharacter(Character character)	82

3.12.4.4.14	importCharacters(String screenplay, Context context)	82
3.12.4.4.15	goToListSpeechesActivity(Context context, String selected)	83
3.12.4.4.16	goToListCharactersActivity(Context context)	83
3.12.4.4.17	goToEditChapterActivity(Context context, String selected)	83
3.12.4.4.18	goToNewChapterActivity(Context context)	84
3.12.4.4.19	goToNewCharacterActivity(Context context)	84
3.12.5	ScreenplayPresenterImpl (class)	85
3.12.5.1	Descrizione	85
3.12.5.2	Utilizzo	85
3.12.5.3	Classi ereditate	85
3.12.5.4	Attributi	85
3.12.5.4.1	newScreenplayInterface	85
3.12.5.4.2	newChapterInterface	86
3.12.5.4.3	screenplay	86
3.12.5.4.4	listChapterInterface	86
3.12.5.4.5	titlesAdapter	86
3.12.5.5	Metodi	86
3.12.5.5.1	ScreenplayPresenterImpl(Screenplay screenplay)	87
3.12.5.5.2	setActivity(ListChapterInterface listChapterInterface)	87
3.12.5.5.3	getScreenplay()	87
3.12.5.5.4	getScreenplayTitle()	87
3.12.5.5.5	getTitlesAdapter(Context context, String screenplay)	87
3.12.5.5.6	goToListSpeechesActivity(Context context, String selected)	88
3.12.5.5.7	goToListCharactersActivity(Context context)	88
3.12.5.5.8	goToEditChapterActivity(Context context, String selected)	88
3.12.5.5.9	goToNewChapterActivity(Context context)	89
3.12.5.5.10	goToNewCharacterActivity(Context context)	89
3.12.5.5.11	export()	89
3.12.5.5.12	share()	89
3.12.5.5.13	newScreenplay(String title, Context context)	90
3.12.5.5.14	addCharacter(Character character)	90
3.12.5.5.15	importCharacters(String screenplay, Context context)	90
3.12.5.5.16	newChapter(String title)	90
3.12.5.5.17	moveUpChapter(String chapterTitle)	91
3.12.5.5.18	moveDownChapter(String chapterTitle)	91
3.12.5.5.19	save(Screenplay screenplay, Context context)	91
3.12.6	ChapterPresenter (interface)	92
3.12.6.1	Descrizione	92
3.12.6.2	Utilizzo	92
3.12.6.3	Classi figlie	92
3.12.6.4	Metodi	92

3.12.6.4.1	moveUpSpeech(Speech speech)	92
3.12.6.4.2	moveDownSpeech(Speech speech)	93
3.12.6.4.3	setActivity(NewSpeechInterface newSpeechInterface)	93
3.12.6.4.4	getSpeeches(Context context)	93
3.12.6.4.5	getCharactersAdapter(Context context)	94
3.12.6.4.6	getChapterTitle()	94
3.12.6.4.7	getSpeechSelected()	94
3.12.6.4.8	setChapterTitle(String title)	94
3.12.6.4.9	setSpeechSelected(int position)	94
3.12.6.4.10	newSpeech(String text, String characterName, String emotion)	95
3.12.6.4.11	deleteSpeech(Speech speech)	95
3.12.6.4.12	goToEditSpeechActivity(Context context, Speech selected)	95
3.12.6.4.13	goToListCharactersActivity(Context context)	95
3.12.6.4.14	goToEditChapterActivity(Context context)	96
3.12.6.4.15	goToNewCharacterActivity(Context context)	96
3.12.6.4.16	goToNewSpeechActivity(Context context)	96
3.12.7	ChapterPresenterImpl (class)	97
3.12.7.1	Descrizione	97
3.12.7.2	Utilizzo	97
3.12.7.3	Classi ereditate	97
3.12.7.4	Attributi	97
3.12.7.4.1	chapter	97
3.12.7.4.2	characterContainer	98
3.12.7.4.3	projectName	98
3.12.7.4.4	speechSelected	98
3.12.7.4.5	speechArrayAdapter	98
3.12.7.4.6	listSpeechesInterface	98
3.12.7.4.7	newChapterInterface	99
3.12.7.4.8	editChapterInterface	99
3.12.7.4.9	newSpeechInterface	99
3.12.7.5	Metodi	99
3.12.7.5.1	ChapterPresenterImpl(Chapter chapter, CharacterContainer characterContainer, String projectName)	99
3.12.7.5.2	setActivity(ListSpeechesInterface listSpeechesInterface)	100
3.12.7.5.3	getChapterTitle()	100
3.12.7.5.4	getSpeeches()	100
3.12.7.5.5	getCharactersAdapter(Context context)	100
3.12.7.5.6	getSpeechSelected()	100
3.12.7.5.7	setChapterTitle(String title)	101
3.12.7.5.8	setSpeechSelected(int position)	101
3.12.7.5.9	newSpeech(String text, String characterName, String emotion)	101
3.12.7.5.10	deleteSpeech(Speech speech)	101

3.12.7.5.11	goToEditSpeechActivity(Context context, Speech selected)	102
3.12.7.5.12	goToListCharactersActivity(Context context)	102
3.12.7.5.13	goToEditChapterActivity(Context context)	102
3.12.7.5.14	goToNewCharacterActivity(Context context)	102
3.12.7.5.15	goToNewSpeechActivity(Context context)	103
3.12.7.5.16	moveUpSpeech(Speech speech)	103
3.12.7.5.17	moveDownSpeech(Speech speech)	103
3.12.8	SpeechPresenter (interface)	104
3.12.8.1	Descrizione	104
3.12.8.2	Utilizzo	104
3.12.8.3	Classi figlie	104
3.12.8.4	Metodi	104
3.12.8.4.1	setActivity(EditSpeechInterface editSpeechInterface)	104
3.12.8.4.2	getSpeechText()	104
3.12.8.4.3	getSpeechEmotion()	105
3.12.8.4.4	getSpeechCharacter()	105
3.12.8.4.5	getScreenplayCharacters()	105
3.12.8.4.6	setSpeechText(String text)	105
3.12.8.4.7	setSpeechEmotion(String emotion)	105
3.12.8.4.8	setSpeechCharacter(Character character)	106
3.12.9	SpeechPresenterImpl (class)	106
3.12.9.1	Descrizione	106
3.12.9.2	Utilizzo	106
3.12.9.3	Classi ereditate	106
3.12.9.4	Attributi	106
3.12.9.4.1	characterContainer	106
3.12.9.4.2	editSpeechInterface	107
3.12.9.4.3	speech	107
3.12.9.5	Metodi	107
3.12.9.5.1	SpeechPresenterImpl(Speech speech, CharacterContainer characterContainer)	107
3.12.9.5.2	SpeechPresenterImpl(EditSpeechInterface editSpeechActivity, Speech speech)	107
3.12.9.5.3	setActivity(EditSpeechInterface editSpeechInterface)	108
3.12.9.5.4	getSpeechText()	108
3.12.9.5.5	getSpeechEmotion()	108
3.12.9.5.6	getScreenplayCharacters()	108
3.12.9.5.7	getSpeechCharacter()	108
3.12.9.5.8	setSpeechText(String text)	108
3.12.9.5.9	setSpeechEmotion(String emotion)	109
3.12.9.5.10	setSpeechCharacter(Character character)	109
3.12.10	CharacterPresenter (interface)	109
3.12.10.1	Descrizione	109
3.12.10.2	Utilizzo	109
3.12.10.3	Classi figlie	110
3.12.10.4	Metodi	110

3.12.10.4.1	setActivity(ListCharacterInterface listCharacterInterface)	110
3.12.10.4.2	setActivity(NewCharacterInterface newCharacterInterface)	110
3.12.10.4.3	setActivity(EditCharacterInterface editCharacterInterface)	110
3.12.10.4.4	getCharacter()	110
3.12.10.4.5	getCharacterArrayAdapter(Context context)	111
3.12.10.4.6	getProjectName()	111
3.12.10.4.7	newCharacter(String name, String voice, Avatar avatar)	111
3.12.10.4.8	goToEditCharacterActivity(Context context, Character character)	111
3.12.10.4.9	goToNewCharacterActivity(Context context)	112
3.12.10.4.10	goToListCharacterActivity(Context context)	112
3.12.11	CharacterPresenterImpl (class)	112
3.12.11.1	Descrizione	112
3.12.11.2	Utilizzo	112
3.12.11.3	Classi ereditate	113
3.12.11.4	Attributi	113
3.12.11.4.1	projectName	113
3.12.11.4.2	characterContainer	113
3.12.11.4.3	character	113
3.12.11.4.4	newCharacterInterface	113
3.12.11.4.5	listCharacterInterface	114
3.12.11.4.6	editCharacterInterface	114
3.12.11.5	Metodi	114
3.12.11.5.1	CharacterPresenterImpl(CharacterContainer characterContainer, String projectName)	114
3.12.11.5.2	CharacterPresenterImpl(Character character, String projectName)	114
3.12.11.5.3	CharacterPresenterImpl(NewCharacterInterface characterInterface)	115
3.12.11.5.4	setActivity(ListCharacterInterface listCharacterInterface)	115
3.12.11.5.5	setActivity(EditCharacterInterface editCharacterInterface)	115
3.12.11.5.6	setActivity(NewCharacterInterface newCharacterInterface)	115
3.12.11.5.7	getCharacterArrayAdapter(Context context)	116
3.12.11.5.8	getCharacter()	116
3.12.11.5.9	getProjectName()	116
3.12.11.5.10	newCharacter(String name, String voice, Avatar avatar)	116
3.12.11.5.11	goToEditCharacterActivity(Context context, Character selected)	117
3.12.11.5.12	goToNewCharacterActivity(Context context)	117
3.12.11.5.13	goToListCharacterActivity(Context context)	117
3.12.12	CharacterArrayAdapter (class)	117

3.12.12.1	Descrizione	117
3.12.12.2	Utilizzo	118
3.12.12.3	Attributi	118
3.12.12.3.1	characterList	118
3.12.12.3.2	characterName	118
3.12.12.3.3	characterAvatar	118
3.12.12.4	Metodi	118
3.12.12.4.1	CharacterAdapter(Context context, int re- source)	118
3.12.12.4.2	add(Character object)	119
3.12.12.4.3	getCount()	119
3.12.12.4.4	getItem(int index)	119
3.12.12.4.5	getView(int position, View convertView, ViewGroup parent)	119
3.12.13	SpeechAdapter (class)	120
3.12.13.1	Descrizione	120
3.12.13.2	Utilizzo	120
3.12.13.3	Attributi	120
3.12.13.3.1	speechList	120
3.12.13.3.2	speechText	120
3.12.13.3.3	speechAvatar	120
3.12.13.3.4	characterName	121
3.12.13.3.5	speechSelected	121
3.12.13.4	Metodi	121
3.12.13.4.1	SpeechAdapter(Context context, int resour- ce)	121
3.12.13.4.2	add(Speech object)	121
3.12.13.4.3	getCount()	121
3.12.13.4.4	getItem(int index)	122
3.12.13.4.5	setSpeechSelected(int position)	122
3.12.13.4.6	getView(int position, View convertView, ViewGroup parent)	122
3.12.14	StringAdapter (class)	122
3.12.14.1	Descrizione	123
3.12.14.2	Utilizzo	123
3.12.14.3	Attributi	123
3.12.14.3.1	textList	123
3.12.14.3.2	textView	123
3.12.14.3.3	stringSelected	123
3.12.14.4	Metodi	123
3.12.14.4.1	StringAdapter(Context context)	123
3.12.14.4.2	add(String object)	124
3.12.14.4.3	getCount()	124
3.12.14.4.4	getItem(int index)	124
3.12.14.4.5	setStringSelected(int position)	124
3.12.14.4.6	getView(int position, View convertView, ViewGroup parent)	125
3.12.15	EmotionAdapter (class)	125
3.12.15.1	Descrizione	125

3.12.15.2	Utilizzo	125
3.12.15.3	Attributi	125
3.12.15.3.1	emotionList	125
3.12.15.3.2	context	126
3.12.15.4	Metodi	126
3.12.15.4.1	EmotionArrayAdapter(Context context)	126
3.12.15.4.2	getItem(int index)	126
3.12.15.4.3	getItemID(int position)	126
3.12.15.4.4	getView(int position, View convertView, ViewGroup parent)	127
3.13	Sivodim::Drama::View	127
3.13.1	Informazioni sul package	127
3.13.2	HomeInterface (interface)	128
3.13.2.1	Descrizione	128
3.13.2.2	Utilizzo	129
3.13.2.3	Classi figlie	129
3.13.3	HomeActivity (class)	129
3.13.3.1	Descrizione	129
3.13.3.2	Utilizzo	129
3.13.3.3	Classi ereditate	129
3.13.3.4	Attributi	129
3.13.3.4.1	homePresenter	129
3.13.3.4.2	screenplayListView	130
3.13.3.4.3	screenplayListAdapter	130
3.13.3.5	Metodi	130
3.13.3.5.1	onCreate(Bundle savedInstanceState)	130
3.13.4	ListChapterInterface (interface)	130
3.13.4.1	Descrizione	130
3.13.4.2	Utilizzo	130
3.13.4.3	Classi figlie	130
3.13.5	ListChapterActivity (class)	131
3.13.5.1	Descrizione	131
3.13.5.2	Utilizzo	131
3.13.5.3	Classi ereditate	131
3.13.5.4	Attributi	131
3.13.5.4.1	screenplayPresenter	131
3.13.5.4.2	chapterListView	131
3.13.5.4.3	chapterListAdapter	132
3.13.5.5	Metodi	132
3.13.5.5.1	onCreate(Bundle savedInstanceState)	132
3.13.6	NewScreenplayInterface (interface)	132
3.13.6.1	Descrizione	132
3.13.6.2	Utilizzo	132
3.13.6.3	Classi figlie	132
3.13.7	NewScreenplayActivity (class)	133
3.13.7.1	Descrizione	133
3.13.7.2	Utilizzo	133
3.13.7.3	Classi ereditate	133
3.13.7.4	Attributi	133

3.13.7.4.1	buttonCreateProject	133
3.13.7.4.2	editTextProjectName	133
3.13.7.4.3	spinnerImportCharacter	134
3.13.7.4.4	screenplayPresenter	134
3.13.7.5	Metodi	134
3.13.7.5.1	onCreate(Bundle savedInstanceState)	134
3.13.8	ListSpeechesInterface (interface)	134
3.13.8.1	Descrizione	134
3.13.8.2	Utilizzo	134
3.13.8.3	Classi figlie	134
3.13.9	NewChapterInterface (interface)	135
3.13.9.1	Descrizione	135
3.13.9.2	Utilizzo	135
3.13.9.3	Classi figlie	135
3.13.10	NewChapterActivity (class)	135
3.13.10.1	Descrizione	135
3.13.10.2	Utilizzo	135
3.13.10.3	Classi ereditate	135
3.13.10.4	Attributi	135
3.13.10.4.1	screenplayPresenter	136
3.13.10.4.2	newChapterName	136
3.13.10.4.3	createNewChapter	136
3.13.10.5	Metodi	136
3.13.10.5.1	onCreate(Bundle savedInstanceState)	136
3.13.11	EditChapterInterface (interface)	136
3.13.11.1	Descrizione	136
3.13.11.2	Utilizzo	137
3.13.11.3	Classi figlie	137
3.13.12	EditChapterActivity (class)	137
3.13.12.1	Descrizione	137
3.13.12.2	Utilizzo	137
3.13.12.3	Classi ereditate	137
3.13.12.4	Attributi	137
3.13.12.4.1	chapterPresenter	137
3.13.12.4.2	chapterName	138
3.13.12.4.3	editSoundtrack	138
3.13.12.4.4	editWallpaper	138
3.13.12.4.5	apply	138
3.13.12.5	Metodi	138
3.13.12.5.1	onCreate(Bundle savedInstanceState)	138
3.13.13	NewSpeechInterface (interface)	139
3.13.13.1	Descrizione	139
3.13.13.2	Utilizzo	139
3.13.13.3	Classi figlie	139
3.13.14	NewSpeechActivity (class)	139
3.13.14.1	Descrizione	139
3.13.14.2	Utilizzo	139
3.13.14.3	Classi ereditate	139
3.13.14.4	Attributi	139

3.13.14.4.1	chapterPresenter	140
3.13.14.4.2	newSpeechText	140
3.13.14.4.3	newSpeechCharacter	140
3.13.14.4.4	newSpeechEmotion	140
3.13.14.4.5	newSpeechApply	140
3.13.14.5	Metodi	140
3.13.14.5.1	onCreate(Bundle savedInstanceState)	141
3.13.15	EditSpeechInterface (interface)	141
3.13.15.1	Descrizione	141
3.13.15.2	Utilizzo	141
3.13.15.3	Classi figlie	141
3.13.16	EditSpeechActivity (class)	141
3.13.16.1	Descrizione	141
3.13.16.2	Utilizzo	141
3.13.16.3	Classi ereditate	142
3.13.16.4	Attributi	142
3.13.16.4.1	speechPresenter	142
3.13.16.4.2	speechText	142
3.13.16.4.3	emotion	142
3.13.16.4.4	character	142
3.13.16.4.5	applyChanges	142
3.13.16.5	Metodi	143
3.13.16.5.1	onCreate(Bundle savedInstanceState)	143
3.13.17	NewCharacterInterface (interface)	143
3.13.17.1	Descrizione	143
3.13.17.2	Utilizzo	143
3.13.17.3	Classi figlie	143
3.13.18	NewCharacterActivity (class)	144
3.13.18.1	Descrizione	144
3.13.18.2	Utilizzo	144
3.13.18.3	Classi ereditate	144
3.13.18.4	Attributi	144
3.13.18.4.1	characterPresenter	144
3.13.18.4.2	newCharacterName	144
3.13.18.4.3	newCharacterVoice	145
3.13.18.4.4	newCharacterAvatar	145
3.13.18.4.5	newCharacterApply	145
3.13.18.4.6	newCharacterTestVoice	145
3.13.18.4.7	avatarPath	145
3.13.18.4.8	myPermissionsRequestReadMemory	146
3.13.18.4.9	resultLoadImage	146
3.13.18.5	Metodi	146
3.13.18.5.1	onCreate(Bundle savedInstanceState)	146
3.13.19	ListCharacterInterface (interface)	146
3.13.19.1	Descrizione	146
3.13.19.2	Utilizzo	146
3.13.19.3	Classi figlie	147
3.13.20	ListCharacterActivity (class)	147
3.13.20.1	Descrizione	147

3.13.20.2 Utilizzo	147
3.13.20.3 Classi ereditate	147
3.13.20.4 Attributi	147
3.13.20.4.1 characterPresenter	147
3.13.20.4.2 characterListView	147
3.13.20.4.3 characterArrayAdapter	148
3.13.20.5 Metodi	148
3.13.20.5.1 onCreate(Bundle savedInstanceState)	148
3.13.21 EditCharacterInterface (interface)	148
3.13.21.1 Descrizione	148
3.13.21.2 Utilizzo	148
3.13.21.3 Classi figlie	148
3.13.22 EditCharacterActivity (class)	149
3.13.22.1 Descrizione	149
3.13.22.2 Utilizzo	149
3.13.22.3 Classi ereditate	149
3.13.22.4 Attributi	149
3.13.22.4.1 characterPresenter	149
3.13.22.4.2 avatarPath	149
3.13.22.4.3 editAvatar	150
3.13.22.4.4 editVoice	150
3.13.22.4.5 editName	150
3.13.22.4.6 apply	150
3.13.22.4.7 character	150
3.13.22.4.8 resultLoadImage	151
3.13.22.4.9 myPermissionsRequestReadMemory	151
3.13.22.5 Metodi	151
3.13.22.5.1 onCreate(Bundle savedInstanceState)	151
3.13.23 ListSpeechesActivity (class)	151
3.13.23.1 Descrizione	151
3.13.23.2 Utilizzo	152
3.13.23.3 Classi ereditate	152
3.13.23.4 Attributi	152
3.13.23.4.1 chapterPresenter	152
3.13.23.4.2 speechListView	152
3.13.23.4.3 speechListAdapter	152
3.13.23.4.4 moveButtons	152
3.13.23.4.5 upButton	153
3.13.23.4.6 downButton	153
3.13.23.4.7 doneButton	153
3.13.23.4.8 deleteButton	153
3.13.23.4.9 addSpeech	153
3.13.23.5 Metodi	153
3.13.23.5.1 ListSpeechesActivity()	154
3.13.23.5.2 setPresenter(ChapterPresenter chapterPresenter)	
	154
3.14 Sivodim::Drama::Model::Utilities	154
3.14.1 Informazioni sul package	154
3.14.2 Sound (abstract)	155

3.14.2.1	Descrizione	155
3.14.2.2	Utilizzo	155
3.14.2.3	Classi figlie	155
3.14.2.4	Attributi	155
3.14.2.4.1	path	155
3.14.2.4.2	audio	155
3.14.2.4.3	mediaplayer	156
3.14.2.4.4	onPause	156
3.14.2.5	Metodi	156
3.14.2.5.1	Sound(String path)	156
3.14.2.5.2	getAudio()	156
3.14.2.5.3	pause()	156
3.14.2.5.4	stop()	157
3.14.2.5.5	play()	157
3.14.2.5.6	maxSize()	157
3.14.2.5.7	isDefined()	157
3.14.2.5.8	getDuration()	157
3.14.3	SoundFx (class)	158
3.14.3.1	Descrizione	158
3.14.3.2	Utilizzo	158
3.14.3.3	Classi ereditate	158
3.14.3.4	Attributi	158
3.14.3.4.1	maxSize	158
3.14.3.5	Metodi	158
3.14.3.5.1	SoundFx(String path)	158
3.14.3.5.2	maxSize()	159
3.14.4	SpeechSound (class)	159
3.14.4.1	Descrizione	159
3.14.4.2	Utilizzo	159
3.14.4.3	Classi ereditate	159
3.14.4.4	Attributi	159
3.14.4.4.1	maxSize	159
3.14.4.5	Metodi	159
3.14.4.5.1	SpeechSound(String path)	160
3.14.4.5.2	maxSize()	160
3.14.5	Soundtrack (class)	160
3.14.5.1	Descrizione	160
3.14.5.2	Utilizzo	160
3.14.5.3	Classi ereditate	160
3.14.5.4	Attributi	160
3.14.5.4.1	maxSize	161
3.14.5.5	Metodi	161
3.14.5.5.1	Soundtrack(String path)	161
3.14.5.5.2	maxSize()	161
3.14.6	Image (abstract)	161
3.14.6.1	Descrizione	161
3.14.6.2	Utilizzo	162
3.14.6.3	Classi figlie	162
3.14.6.4	Attributi	162

3.14.6.4.1	path	162
3.14.6.4.2	image	162
3.14.6.5	Metodi	162
3.14.6.5.1	Image(String path)	162
3.14.6.5.2	maxSize()	163
3.14.7	Background (class)	163
3.14.7.1	Descrizione	163
3.14.7.2	Utilizzo	163
3.14.7.3	Classi ereditate	163
3.14.7.4	Attributi	163
3.14.7.4.1	maxSize	163
3.14.7.5	Metodi	163
3.14.7.5.1	Background(String path)	164
3.14.7.5.2	maxSize()	164
3.14.8	Avatar (class)	164
3.14.8.1	Descrizione	164
3.14.8.2	Utilizzo	164
3.14.8.3	Classi ereditate	164
3.14.8.4	Attributi	164
3.14.8.4.1	maxSize	164
3.14.8.5	Metodi	165
3.14.8.5.1	Avatar(String path)	165
3.14.8.5.2	maxSize()	165
3.14.9	XMLParser (class)	165
3.14.9.1	Descrizione	165
3.14.9.2	Utilizzo	165
3.14.9.3	Attributi	165
3.14.9.3.1	parsedData	166
3.14.9.4	Metodi	166
3.14.9.4.1	XMLParser()	166
3.14.9.4.2	getParsedData()	166
3.14.9.4.3	parseCharacter(Node node)	166
3.14.9.4.4	parseChapter(Node node)	166
3.14.9.4.5	parseXml(File file)	167
3.14.9.4.6	saveXML(File file, Screenplay screenplay)	167
3.15	Sivodim::Drama::Model::Chapter	167
3.15.1	Informazioni sul package	167
3.15.2	Chapter (interface)	168
3.15.2.1	Descrizione	168
3.15.2.2	Utilizzo	168
3.15.2.3	Classi figlie	169
3.15.2.4	Metodi	169
3.15.2.4.1	getSpeechIterator()	169
3.15.2.4.2	getTitle()	169
3.15.2.4.3	setSoundtrack(Soundtrack soundtrack)	169
3.15.2.4.4	setTitle(String title)	169
3.15.2.4.5	setBackground(Background background)	170
3.15.2.4.6	addSpeech(Speech speech)	170
3.15.2.4.7	deleteSpeech(Speech speech)	170

3.15.2.4.8	deleteBackground()	170
3.15.2.4.9	deleteSoundtrack()	170
3.15.2.4.10	moveUpSpeech(ListIterator<Speech> iterator)	171
3.15.2.4.11	moveDownSpeech(ListIterator<Speech> iterator)	171
3.15.2.4.12	getDuration()	171
3.15.2.4.13	getSoundtrack()	171
3.15.2.4.14	getBackground()	171
3.15.3	ChapterImpl (class)	172
3.15.3.1	Descrizione	172
3.15.3.2	Utilizzo	172
3.15.3.3	Classi ereditate	172
3.15.3.4	Attributi	172
3.15.3.4.1	title	172
3.15.3.4.2	background	172
3.15.3.4.3	soundtrack	173
3.15.3.4.4	soundFx	173
3.15.3.4.5	speeches	173
3.15.3.5	Metodi	173
3.15.3.5.1	getTitle()	173
3.15.3.5.2	ChapterImpl(ChapterBuilder builder)	173
3.15.3.5.3	getSpeechIterator()	174
3.15.3.5.4	moveUpSpeech(ListIterator<Speech> iterator)	174
3.15.3.5.5	moveDownSpeech(ListIterator<Speech> iterator)	174
3.15.3.5.6	addSpeech(Speech speech)	174
3.15.3.5.7	setTitle(String title)	175
3.15.3.5.8	setBackground(Background background)	175
3.15.3.5.9	setSoundtrack(Soundtrack soundtrack)	175
3.15.3.5.10	deleteSpeech(Speech speech)	175
3.15.3.5.11	deleteBackground()	176
3.15.3.5.12	deleteSoundtrack()	176
3.15.3.5.13	getDuration()	176
3.15.3.5.14	getSoundtrack()	176
3.15.3.5.15	getBackground()	176
3.15.4	ChapterBuilder (class)	177
3.15.4.1	Descrizione	177
3.15.4.2	Utilizzo	177
3.15.4.3	Classi ereditate	177
3.15.4.4	Attributi	177
3.15.4.4.1	titleB	177
3.15.4.4.2	backgroundB	177
3.15.4.4.3	soundtrackB	178
3.15.4.4.4	speechesB	178
3.15.4.4.5	soundFxB	178
3.15.4.5	Metodi	178
3.15.4.5.1	ChapterBuilder()	178
3.15.4.5.2	build()	178
3.15.4.5.3	setTitle(String title)	178

3.15.4.5.4	setBackground(Background background)	179
3.15.4.5.5	setSoundtrack(Soundtrack soundtrack)	179
3.15.4.5.6	setSoundFx(SoundFx soundFx)	179
3.15.5	Speech (interface)	180
3.15.5.1	Descrizione	180
3.15.5.2	Utilizzo	180
3.15.5.3	Classi figlie	180
3.15.5.4	Metodi	180
3.15.5.4.1	setText(String text)	180
3.15.5.4.2	setEmotion(String emotionID)	180
3.15.5.4.3	setCharacter(Character character)	181
3.15.5.4.4	setSoundFx(SoundFx soundFx)	181
3.15.5.4.5	getText()	181
3.15.5.4.6	getEmotion()	181
3.15.5.4.7	getCharacter()	182
3.15.5.4.8	getAudioPath()	182
3.15.5.4.9	setAudioPath(String path)	182
3.15.5.4.10	getAudioStatus()	182
3.15.5.4.11	setAudioStatus(boolean status)	182
3.15.5.4.12	synthesizeAudio(Context context, String path) . .	183
3.15.5.4.13	getDuration()	183
3.15.6	SpeechImpl (class)	183
3.15.6.1	Descrizione	183
3.15.6.2	Utilizzo	183
3.15.6.3	Classi ereditate	184
3.15.6.4	Attributi	184
3.15.6.4.1	text	184
3.15.6.4.2	emotionID	184
3.15.6.4.3	Character	184
3.15.6.4.4	soundFx	184
3.15.6.4.5	audioPath	184
3.15.6.4.6	audioStatus	185
3.15.6.5	Metodi	185
3.15.6.5.1	SpeechImpl(SpeechBuilder builder)	185
3.15.6.5.2	setText(String text)	185
3.15.6.5.3	setEmotion(String emotionID)	185
3.15.6.5.4	setCharacter(Character character)	186
3.15.6.5.5	setSoundFx(SoundFx soundFx)	186
3.15.6.5.6	getText()	186
3.15.6.5.7	getEmotion()	186
3.15.6.5.8	getCharacter()	186
3.15.6.5.9	getAudioPath()	187
3.15.6.5.10	getAudioStatus()	187
3.15.6.5.11	getDuration()	187
3.15.6.5.12	setAudioStatus(boolean status)	187
3.15.6.5.13	setAudioPath(String path)	187
3.15.6.5.14	getEmotions(Context context)	187
3.15.6.5.15	getVoices(Context context)	188
3.15.6.5.16	synthesizeAudio(Context context, String path) . .	188

3.15.7	SpeechBuilder (class)	188
3.15.7.1	Descrizione	188
3.15.7.2	Utilizzo	188
3.15.7.3	Attributi	189
3.15.7.3.1	audioPathB	189
3.15.7.3.2	audioStatusB	189
3.15.7.3.3	textB	189
3.15.7.3.4	emotionIDB	189
3.15.7.3.5	characterB	189
3.15.7.3.6	soundFxB	190
3.15.7.4	Metodi	190
3.15.7.4.1	SpeechBuilder()	190
3.15.7.4.2	setText(String text)	190
3.15.7.4.3	setEmotion(String emotionID)	190
3.15.7.4.4	setCharacter(Character character)	191
3.15.7.4.5	setSoundFX(SoundFx soundFx)	191
3.15.7.4.6	build()	191
3.15.7.4.7	setAudioPath(String path)	191
3.15.7.4.8	setAudioStatus(boolean status)	192
3.16	Sivodim::Drama::Model::Character	192
3.16.1	Informazioni sul package	192
3.16.2	Character (interface)	192
3.16.2.1	Descrizione	192
3.16.2.2	Utilizzo	193
3.16.2.3	Classi figlie	193
3.16.2.4	Metodi	193
3.16.2.4.1	clone()	193
3.16.2.4.2	getAvatar()	193
3.16.2.4.3	getName()	193
3.16.2.4.4	getVoiceld()	193
3.16.2.4.5	setAvatar(Avatar avatar)	194
3.16.2.4.6	setName(String name)	194
3.16.2.4.7	setVoice(String voicelD)	194
3.16.3	CharacterImpl (class)	195
3.16.3.1	Descrizione	195
3.16.3.2	Utilizzo	195
3.16.3.3	Classi ereditate	195
3.16.3.4	Attributi	195
3.16.3.4.1	avatar	195
3.16.3.4.2	name	195
3.16.3.4.3	voicelD	196
3.16.3.4.4	name	196
3.16.3.4.5	avatar	196
3.16.3.4.6	voicelD	196
3.16.3.5	Metodi	196
3.16.3.5.1	CharacterImpl(CharacterBuilder builder)	196
3.16.3.5.2	clone()	197
3.16.3.5.3	setAvatar(Avatar avatar)	197
3.16.3.5.4	setVoice(String voicelD)	197

3.16.3.5.5	setName(String name)	197
3.16.3.5.6	getName()	198
3.16.3.5.7	getAvatar()	198
3.16.3.5.8	getVoiceld()	198
3.16.4	CharacterBuilder (class)	198
3.16.4.1	Descrizione	198
3.16.4.2	Utilizzo	198
3.16.4.3	Classi ereditate	199
3.16.4.4	Metodi	199
3.16.4.4.1	CharacterBuilder()	199
3.16.4.4.2	setName(String name)	199
3.16.4.4.3	setAvatar(Avatar avatar)	199
3.16.4.4.4	setVoice(String voiceID)	199
3.16.5	CharacterContainer (class)	200
3.16.5.1	Descrizione	200
3.16.5.2	Utilizzo	200
3.16.5.3	Attributi	200
3.16.5.3.1	characters	200
3.16.5.4	Metodi	200
3.16.5.4.1	CharacterContainer()	200
3.16.5.4.2	CharacterContainer(ArrayList<Character> cha- racters)	200
3.16.5.4.3	cloneList()	201
3.16.5.4.4	clone()	201
3.16.5.4.5	iterator()	201
3.16.5.4.6	addCharacter(Character character)	201
3.16.5.4.7	removeCharacter(Character character)	201
3.16.5.4.8	getCharacterByName(String name)	202
3.17	Sivodim::Drama::Model::Screenplay	202
3.17.1	Informazioni sul package	202
3.17.2	Screenplay (interface)	204
3.17.2.1	Descrizione	204
3.17.2.2	Utilizzo	204
3.17.2.3	Classi figlie	204
3.17.2.4	Metodi	204
3.17.2.4.1	addChapter(Chapter chapter)	204
3.17.2.4.2	addCharacter(Character character)	205
3.17.2.4.3	export(String type, Context context, TextView feedback)	205
3.17.2.4.4	getChapter(String title)	205
3.17.2.4.5	getChapterIterator()	205
3.17.2.4.6	getCharacterByName(String name)	206
3.17.2.4.7	getCharacters()	206
3.17.2.4.8	getTitle()	206
3.17.2.4.9	importCharacters(Screenplay screenplay)	206
3.17.2.4.10	removeCharacter(Character character)	207
3.17.2.4.11	removeChapter(Chapter chapter)	207
3.17.2.4.12	moveUpChapter(int index)	207
3.17.2.4.13	moveDownChapter(int index)	207

3.17.2.4.14	getPath()	208
3.17.2.4.15	getNextSpeechId()	208
3.17.3	ScreenplayImpl (class)	208
3.17.3.1	Descrizione	208
3.17.3.2	Utilizzo	208
3.17.3.3	Classi ereditate	209
3.17.3.4	Attributi	209
3.17.3.4.1	chapters	209
3.17.3.4.2	characterContainer	209
3.17.3.4.3	exportAlgorithm	209
3.17.3.4.4	nextSpeechId	209
3.17.3.4.5	title	209
3.17.3.5	Metodi	210
3.17.3.5.1	ScreenplayImpl(String title, int nextSpeechId)	210
3.17.3.5.2	loadScreenplay(String screenplayTitle, Context context)	210
3.17.3.5.3	saveScreenplay(Screenplay screenplay, Context context)	210
3.17.3.5.4	addChapter(Chapter chapter)	211
3.17.3.5.5	addCharacter(Character character)	211
3.17.3.5.6	export(String type, Context context, TextView feedback)	211
3.17.3.5.7	getChapter(String title)	211
3.17.3.5.8	getChapterIterator()	212
3.17.3.5.9	getCharacterByName(String name)	212
3.17.3.5.10	getCharacters()	212
3.17.3.5.11	getTitle()	212
3.17.3.5.12	importCharacters(Screenplay screenplay)	213
3.17.3.5.13	removeCharacter(Character character)	213
3.17.3.5.14	removeChapter(Chapter chapter)	213
3.17.3.5.15	moveUpChapter(int index)	213
3.17.3.5.16	moveDownChapter(int index)	214
3.17.3.5.17	getPath()	214
3.17.3.5.18	getNextSpeechId()	214
3.17.4	FfmpegConnector (abstract)	214
3.17.4.1	Descrizione	214
3.17.4.2	Utilizzo	214
3.17.4.3	Classi figlie	215
3.17.4.4	Attributi	215
3.17.4.4.1	ffmpeg	215
3.17.4.4.2	context	215
3.17.4.4.3	errors	215
3.17.4.4.4	finish	215
3.17.4.5	Metodi	215
3.17.4.5.1	FfmpegConnector(Context context)	216
3.17.4.5.2	exec()	216
3.17.4.5.3	getCommand()	216
3.17.5	ExportAlgorithm (abstract)	216
3.17.5.1	Descrizione	216

3.17.5.2	Utilizzo	216
3.17.5.3	Classi figlie	216
3.17.5.4	Attributi	217
3.17.5.4.1	screenplay	217
3.17.5.5	Metodi	217
3.17.5.5.1	export(Context context)	217
3.17.5.5.2	ExportAlgorithm()	217
3.17.5.5.3	setScreenplay(Screenplay screenplay)	217
3.17.6	AudioExport (class)	218
3.17.6.1	Descrizione	218
3.17.6.2	Utilizzo	218
3.17.6.3	Classi ereditate	218
3.17.6.4	Classi figlie	218
3.17.6.5	Attributi	218
3.17.6.5.1	feedback	218
3.17.6.6	Metodi	218
3.17.6.6.1	AudioExport(TextView feedback)	218
3.17.6.6.2	concatenateSpeeches()	219
3.17.6.6.3	addSoundtrack()	219
3.17.6.6.4	finalizeExport(Context context)	219
3.17.6.6.5	export(Context context)	219
3.17.7	ExportPreview (class)	219
3.17.7.1	Descrizione	219
3.17.7.2	Utilizzo	220
3.17.7.3	Classi ereditate	220
3.17.7.4	Attributi	220
3.17.7.4.1	speechSound	220
3.17.7.5	Metodi	220
3.17.7.5.1	ExportPreview(TextView textView, SpeechSound speechSound)	220
3.17.7.5.2	finalizeExport(Context context)	220
3.17.8	VideoExport (class)	221
3.17.8.1	Descrizione	221
3.17.8.2	Utilizzo	221
3.17.8.3	Classi ereditate	221
3.17.8.4	Attributi	221
3.17.8.4.1	audioExport	221
3.17.8.4.2	feedback	221
3.17.8.4.3	mediaTypeImage	222
3.17.8.4.4	mediaTypeVideo	222
3.17.8.4.5	timePassed	222
3.17.8.5	Metodi	222
3.17.8.5.1	VideoExport(TextView feedback)	222
3.17.8.5.2	exportAudio(Context context)	222
3.17.8.5.3	concatenateChapters(Context context, ArrayList<File> chapterExporters)	223
3.17.8.5.4	finalizeExport()	223
3.17.8.5.5	export(Context context)	223
3.17.8.5.6	createOverlay()	223

3.17.8.5.7	makeChapterVideo()	223
3.17.8.5.8	getOutputMediaFile()	224
3.17.8.5.9	getOutputMediaFileUri()	224
3.17.9	Mp3Converter (class)	224
3.17.9.1	Descrizione	224
3.17.9.2	Utilizzo	224
3.17.9.3	Classi ereditate	224
3.17.9.4	Attributi	224
3.17.9.4.1	file	224
3.17.9.4.2	destination	225
3.17.9.5	Metodi	225
3.17.9.5.1	Mp3Converter(Context context, File fileToConvert, File destination)	225
3.17.9.5.2	setFile(File fileToConvert)	225
3.17.9.5.3	setDestination(File destination)	225
3.17.9.5.4	getCommand()	226
3.17.10	AudioConcatenator (class)	226
3.17.10.1	Descrizione	226
3.17.10.2	Utilizzo	226
3.17.10.3	Classi ereditate	226
3.17.10.4	Attributi	226
3.17.10.4.1	files	226
3.17.10.4.2	destination	227
3.17.10.4.3	listTxt	227
3.17.10.5	Metodi	227
3.17.10.5.1	AudioConcatenator(Context context, File destination)	227
3.17.10.5.2	AudioConcatenator(Context context, Vector<File> files, File destination)	227
3.17.10.5.3	addFile(File file)	228
3.17.10.5.4	setDestination(File destination)	228
3.17.10.5.5	createFileList()	228
3.17.10.5.6	getCommand()	228
3.17.11	AudioMixer (class)	229
3.17.11.1	Descrizione	229
3.17.11.2	Utilizzo	229
3.17.11.3	Classi ereditate	229
3.17.11.4	Attributi	229
3.17.11.4.1	primaryFile	229
3.17.11.4.2	secondaryFile	229
3.17.11.4.3	destination	230
3.17.11.5	Metodi	230
3.17.11.5.1	AudioMixer(Context context, File primaryFile, File secondaryFile, File destination)	230
3.17.11.5.2	setPrimaryFile(File primaryFile)	230
3.17.11.5.3	setSecondaryFile(File secondaryFile)	230
3.17.11.5.4	setDestination(File destination)	231
3.17.11.5.5	getCommand()	231
3.17.12	VideoConcatenator (class)	231

3.17.12.1	Descrizione	231
3.17.12.2	Utilizzo	231
3.17.12.3	Classi ereditate	231
3.17.12.4	Attributi	231
3.17.12.4.1	files	232
3.17.12.4.2	destination	232
3.17.12.4.3	listTxt	232
3.17.12.5	Metodi	232
3.17.12.5.1	VideoConcatenator(Context context, File destination)	232
3.17.12.5.2	add(File file)	232
3.17.12.5.3	createFileList()	233
3.17.12.5.4	getCommand()	233
3.17.13	VideoOverlayer (class)	233
3.17.13.1	Descrizione	233
3.17.13.2	Utilizzo	233
3.17.13.3	Classi ereditate	233
3.17.13.4	Attributi	233
3.17.13.4.1	video	234
3.17.13.4.2	destination	234
3.17.13.4.3	image	234
3.17.13.4.4	begin	234
3.17.13.4.5	end	234
3.17.13.5	Metodi	234
3.17.13.5.1	VideoOverlayer(Context context, File video, Avatar image, int begin, int end, File destination)	235
3.17.13.5.2	setImage(File image)	235
3.17.13.5.3	setBegin(int begin)	235
3.17.13.5.4	setEnd(int end)	235
3.17.13.5.5	setDestination(File destination)	236
3.17.13.5.6	getCommand()	236
3.17.14	ImageVideoConverter (class)	236
3.17.14.1	Descrizione	236
3.17.14.2	Utilizzo	236
3.17.14.3	Classi ereditate	236
3.17.14.4	Attributi	236
3.17.14.4.1	duration	237
3.17.14.4.2	destination	237
3.17.14.4.3	image	237
3.17.14.5	Metodi	237
3.17.14.5.1	ImageVideoConverter(Context context, Background background, int duration, File destination)	237
3.17.14.5.2	getCommand()	237
3.17.15	AudioVideoMixer (class)	238
3.17.15.1	Descrizione	238
3.17.15.2	Utilizzo	238
3.17.15.3	Classi ereditate	238
3.17.15.4	Attributi	238
3.17.15.4.1	audio	238

3.17.15.4.2	destination	238
3.17.15.4.3	video	238
3.17.15.5	Metodi	239
3.17.15.5.1	AudioVideoMixer(Context context, File video, File audio, File destination)	239
3.17.15.5.2	getCommand()	239
4	Diagrammi di sequenza	240
4.1	Creazione sceneggiato	240
4.2	Creazione capitolo	241
4.3	Creazione personaggio	243
4.4	Creazione battuta	244
4.5	Visualizzazione battute	245
4.6	Modifica battuta	246
4.7	Creazione voce	247
4.8	Modifica voce	248
4.9	Richiesta di sintesi vocale	249
A	Tracciamento	249
A.1	Tracciamento Requisiti - Classi	249
A.2	Tracciamento Classi - Requisiti	265
A.3	Tracciamento Modulo - Test	281

Elenco delle tabelle

2	Tracciamento Requisiti - Classi	264
3	Tracciamento Classi - Requisiti	280
4	Tracciamento Modulo - Test	305

Elenco delle figure

1	Componente Sivodim	4
2	Componente Sivodim::Libraries	5
3	Componente Sivodim::Libraries::Model	5
4	Componente Sivodim::Libraries::Model::EngineManager	6
5	Classe Sivodim::Libraries::Model::EngineManager::Engine	6
6	Classe Sivodim::Libraries::Model::EngineManager::Engine.Listener	9
7	Classe Sivodim::Libraries::Model::EngineManager::EngineImpl	10
8	Componente Sivodim::Libraries::Model::Integration	14
9	Classe Sivodim::Libraries::Model::Integration::CheckVoiceData	14
10	Classe Sivodim::Libraries::Model::Integration::DownloadVoiceData	15
11	Classe Sivodim::Libraries::Model::Integration::GetsampleText	16
12	Componente Sivodim::Libraries::Model::Mivoq	18
13	Classe Sivodim::Libraries::Model::Mivoq::AbstractFactory	18
14	Classe Sivodim::Libraries::Model::Mivoq::MivoqConnectionFactory	19
15	Classe Sivodim::Libraries::Model::Mivoq::MivoqConnection	20
16	Classe Sivodim::Libraries::Model::Mivoq::MivoqConnectionImpl	22
17	Classe Sivodim::Libraries::Model::Mivoq::MivoqTTSSingleton	27
18	Classe Sivodim::Libraries::Model::Mivoq::MivoqTTSService	30
19	Classe Sivodim::Libraries::Model::Mivoq::AudioRequest	33
20	Classe Sivodim::Libraries::Model::Mivoq::AuthAudioRequest	35
21	Componente Sivodim::Libraries::Model::Voice	38
22	Classe Sivodim::Libraries::Model::Voice::MivoqVoice	39
23	Classe Sivodim::Libraries::Model::Voice::Language	44
24	Classe Sivodim::Libraries::Model::Voice::Emotion	46
25	Classe Sivodim::Libraries::Model::Voice::Effect	48
26	Classe Sivodim::Libraries::Model::Voice::EffectImpl	50
27	Componente Sivodim::Libraries::Presenter	52
28	Classe Sivodim::Libraries::Presenter::HomePresenter	52
29	Classe Sivodim::Libraries::Presenter::HomePresenterImpl	53
30	Classe Sivodim::Libraries::Presenter::VoicePresenter	55
31	Classe Sivodim::Libraries::Presenter::VoicePresenterImpl	57
32	Classe Sivodim::Libraries::Presenter::VoiceListPresenter	61
33	Classe Sivodim::Libraries::Presenter::VoiceListPresenterImpl	63
34	Componente Sivodim::Libraries::View	66
35	Classe Sivodim::Libraries::View::HomeActivityInterface	66
36	Classe Sivodim::Libraries::View::HomeActivity	67
37	Classe Sivodim::Libraries::View::NewVoiceActivityInterface	67
38	Classe Sivodim::Libraries::View::NewVoiceActivity	68
39	Classe Sivodim::Libraries::View::EditVoiceActivityInterface	69
40	Classe Sivodim::Libraries::View::EditVoiceActivity	70
41	Classe Sivodim::Libraries::View::VoiceListActivityInterface	71
42	Classe Sivodim::Libraries::View::VoiceListActivity	71
43	Componente Sivodim::Drama	73
44	Componente Sivodim::Drama::Model	74
45	Componente Sivodim::Drama::Presenter	75
46	Classe Sivodim::Drama::Presenter::HomePresenter	75
47	Classe Sivodim::Drama::Presenter::HomePresenterImpl	77

48	Classe Sivodim::Drama::Presenter::ScreenplayPresenter	79
49	Classe Sivodim::Drama::Presenter::ScreenplayPresenterImpl	85
50	Classe Sivodim::Drama::Presenter::ChapterPresenter	92
51	Classe Sivodim::Drama::Presenter::ChapterPresenterImpl	97
52	Classe Sivodim::Drama::Presenter::SpeechPresenter	104
53	Classe Sivodim::Drama::Presenter::SpeechPresenterImpl	106
54	Classe Sivodim::Drama::Presenter::CharacterPresenter	109
55	Classe Sivodim::Drama::Presenter::CharacterPresenterImpl	112
56	Classe Sivodim::Drama::Presenter::CharacterAdapter	117
57	Classe Sivodim::Drama::Presenter::SpeechAdapter	120
58	Classe Sivodim::Drama::Presenter::StringAdapter	122
59	Classe Sivodim::Drama::Presenter::EmotionAdapter	125
60	Componente Sivodim::Drama::View	128
61	Classe Sivodim::Drama::View::HomeInterface	128
62	Classe Sivodim::Drama::View::HomeActivity	129
63	Classe Sivodim::Drama::View::ListChapterInterface	130
64	Classe Sivodim::Drama::View::ListChapterActivity	131
65	Classe Sivodim::Drama::View::NewScreenplayInterface	132
66	Classe Sivodim::Drama::View::NewScreenplayActivity	133
67	Classe Sivodim::Drama::View::ListSpeechesInterface	134
68	Classe Sivodim::Drama::View::NewChapterInterface	135
69	Classe Sivodim::Drama::View::NewChapterActivity	135
70	Classe Sivodim::Drama::View::EditChapterInterface	136
71	Classe Sivodim::Drama::View::EditChapterActivity	137
72	Classe Sivodim::Drama::View::NewSpeechInterface	139
73	Classe Sivodim::Drama::View::NewSpeechActivity	139
74	Classe Sivodim::Drama::View::EditSpeechInterface	141
75	Classe Sivodim::Drama::View::EditSpeechActivity	141
76	Classe Sivodim::Drama::View::NewCharacterInterface	143
77	Classe Sivodim::Drama::View::NewCharacterActivity	144
78	Classe Sivodim::Drama::View::ListCharacterInterface	146
79	Classe Sivodim::Drama::View::ListCharacterActivity	147
80	Classe Sivodim::Drama::View::EditCharacterInterface	148
81	Classe Sivodim::Drama::View::EditCharacterActivity	149
82	Classe Sivodim::Drama::View::ListSpeechesActivity	151
83	Componente Sivodim::Drama::Model::Utilities	154
84	Classe Sivodim::Drama::Model::Utilities::Sound	155
85	Classe Sivodim::Drama::Model::Utilities::SoundFx	158
86	Classe Sivodim::Drama::Model::Utilities::SpeechSound	159
87	Classe Sivodim::Drama::Model::Utilities::Soundtrack	160
88	Classe Sivodim::Drama::Model::Utilities::Image	161
89	Classe Sivodim::Drama::Model::Utilities::Background	163
90	Classe Sivodim::Drama::Model::Utilities::Avatar	164
91	Classe Sivodim::Drama::Model::Utilities::XMLParser	165
92	Componente Sivodim::Drama::Model::Chapter	168
93	Classe Sivodim::Drama::Model::Chapter::Chapter	168
94	Classe Sivodim::Drama::Model::Chapter::ChapterImpl	172
95	Classe Sivodim::Drama::Model::Chapter::ChapterBuilder	177
96	Classe Sivodim::Drama::Model::Chapter::Speech	180

97	Classe Sivodim::Drama::Model::Chapter::SpeechImpl	183
98	Classe Sivodim::Drama::Model::Chapter::SpeechBuilder	188
99	Componente Sivodim::Drama::Model::Character	192
100	Classe Sivodim::Drama::Model::Character::Character	192
101	Classe Sivodim::Drama::Model::Character::CharacterImpl	195
102	Classe Sivodim::Drama::Model::Character::CharacterBuilder	198
103	Classe Sivodim::Drama::Model::Character::CharacterContainer	200
104	Componente Sivodim::Drama::Model::Screenplay	203
105	Classe Sivodim::Drama::Model::Screenplay::Screenplay	204
106	Classe Sivodim::Drama::Model::Screenplay::ScreenplayImpl	208
107	Classe Sivodim::Drama::Presenter::FfmpegConnector	214
108	Classe Sivodim::Drama::Model::Screenplay::ExportAlgorithm	216
109	Classe Sivodim::Drama::Model::Screenplay::AudioExport	218
110	Classe Sivodim::Drama::Model::Screenplay::ExportPreview	219
111	Classe Sivodim::Drama::Model::Screenplay::VideoExport	221
112	Classe Sivodim::Drama::Model::Screenplay::Mp3Converter	224
113	Classe Sivodim::Drama::Model::Screenplay::AudioConcatenator	226
114	Classe Sivodim::Drama::Model::Screenplay::AudioMixer	229
115	Classe Sivodim::Drama::Model::Screenplay::VideoConcatenator	231
116	Classe Sivodim::Drama::Model::Screenplay::VideoOverlayer	233
117	Classe Sivodim::Drama::Model::Screenplay::ImageVideoConverter	236
118	Classe Sivodim::Drama::Model::Screenplay::AudioVideoMixer	238
119	Diagramma di sequenza per la creazione di uno sceneggiato	240
120	Diagramma di sequenza per la creazione di un capitolo	241
121	Diagramma di sequenza per la creazione di un personaggio	243
122	Diagramma di sequenza per la creazione di una battuta	244
123	Diagramma di sequenza per la visualizzazione di un insieme di battute	245
124	Diagramma di sequenza per la modifica di una battuta	246
125	Diagramma di sequenza per la creazione di una voce	247
126	Diagramma di sequenza per la modifica di una voce	248
127	Diagramma di sequenza per la richiesta di sintesi vocale	249

1 Introduzione

1.1 Scopo del documento

Questo documento definisce la **progettazione di dettaglio** che i membri del gruppo Stark Labs hanno stilato per la realizzazione del progetto SiVoDiM: Sintesi Vocale per Dispositivi Mobili. Il documento ha lo scopo di ampliare l'architettura del software a partire dai contenuti redatti in *Specifiche Tecniche v2.0.0*. A seguire, viene presentato il design ai *Programmatori* in modo dettagliato e completo, al fine di facilitare la realizzazione del prodotto *software* durante le attività di codifica.

1.2 Scopo del progetto

Lo scopo del progetto risiede nello sviluppo di un'applicazione utile a dimostrare efficacemente le potenzialità del motore di sintesi vocale FA-TTS_G, realizzato dall'azienda MIVOQ s.r.l. e messo a disposizione del gruppo di lavoro. Si devono realizzare due applicazioni per sistemi Android_G:

- **Applicazione di configurazione:** deve permettere all'utente di interfacciarsi direttamente con il sistema operativo per configurare, salvare e modificare le voci ereditate dal motore di sintesi FA-TTS di MIVOQ;
- **Applicazione per la creazione di sceneggiati:** permette la creazione e il salvataggio di racconti e sceneggiati, che possono essere esportati in formato audio attraverso l'utilizzo del motore FA-TTS.

Entrambe le applicazioni devono interfacciarsi con un modulo di basso livello:

- **Modulo di sistema:** permette di interfacciarsi tramite connessione di rete al motore FA-TTS. Fornisce una libreria contenente tutte le funzionalità offerte da FA-TTS, utile nell'ottica di un riuso futuro del *software*.

Lo sviluppo di tutte e quattro le suddette componenti è a carico del gruppo Stark Labs.

1.3 Glossario

Al fine di aumentare la comprensione del testo ed evitare eventuali ambiguità, viene fornito un glossario (*Glossario v3.0.0*) contenente le definizioni degli acronimi e dei termini tecnici utilizzati nei documenti. Ogni vocabolo che ha un riferimento contenuto nel glossario è contrassegnato dal pedice "_G".

1.4 Riferimenti

1.4.1 Normativi

- *Norme di Progetto v3.0.0*;
- *Analisi dei Requisiti v2.0.0*;
- *Specifiche Tecniche v2.0.0*.

1.4.2 Informativi

- Glossario v3.0.0;
- Slide dell'insegnamento Ingegneria del Software modulo A & B (*Design pattern_G*):
<http://www.math.unipd.it/~tullio/IS-1/2015/>
<http://www.math.unipd.it/~rcardin/sweb.html>;
- Documentazione per Android Studio e sviluppo su Android_G:
<http://developer.android.com/tools/studio/index.html>
<http://developer.android.com/guide/index.html>;
- Documentazione Java 7:
<http://docs.oracle.com/javase/7/docs/>;
- Sito ufficiale di FFmpeg:
<https://ffmpeg.org/>;
- Documentazione Volley su Android:
<http://developer.android.com/training/volley/index.html>;
- Documentazione TextToSpeech su Android:
<http://developer.android.com/reference/android/speech/tts/TextToSpeech.html>;
- Documentazione TextToSpeechService su Android:
<http://developer.android.com/reference/android/speech/tts/TextToSpeechService.html>;
- Documentazione SynthesisRequest su Android:
<http://developer.android.com/reference/android/speech/tts/SynthesisRequest.html>.

2 Standard di progetto

2.1 Standard per la progettazione

Gli standard per la progettazione architettonale sono trattati nel documento *Specifica Tecnica v2.0.0*. In particolare, progettazione architettonale e progettazione di dettaglio sono definite rispettivamente nelle sezioni 2.2.1.2 e 2.2.1.3 del documento *Norme di Progetto v3.0.0*. Maggiori dettagli sono riportati nelle sezioni 2.2.2.7 - 2.2.2.10 del medesimo documento.

2.2 Standard per la documentazione del codice

Gli standard per la scrittura della documentazione del codice sono definiti nel documento *Norme di Progetto v3.0.0*.

2.3 Standard di denominazione di entità e relazioni

Gli standard per la denominazione di entità e relazioni sono definiti nel documento *Norme di Progetto v3.0.0*. In particolare, è importante associare a $package_G$, classi, metodi e attributi un nome chiaro e descrittivo che ne rispecchi le funzionalità o il ruolo. Pertanto, è preferibile una denominazione lunga ma esplicativa, piuttosto che sintetica ma ambigua.

2.4 Standard di programmazione

Gli standard di programmazione sono definiti e descritti nel documento *Norme di Progetto v3.0.0*.

2.5 Strumenti di lavoro

Gli strumenti di lavoro da adottare e le rispettive modalità d'uso per un corretto sviluppo del prodotto *software* sono descritti nel documento *Norme di Progetto v3.0.0*.

In particolare, si riportano le tecnologie fondamentali per lo sviluppo del progetto:

- $Android_G$
- Android Studio;
- Java;
- Volley;
- FFmpeg;
- TextToSpeech;
- TextToSpeechService.

Per maggiori dettagli sulle tecnologie di veda la sezione 2 Tecnologie Utilizzate nel documento *Specifica Tecnica v2.0.0*.

3 Specifica componenti

3.1 Sivodim

3.1.1 Informazioni sul package

Di seguito viene illustrata, tramite un diagramma delle classi, la visione di insieme del progetto SiVoDiM.

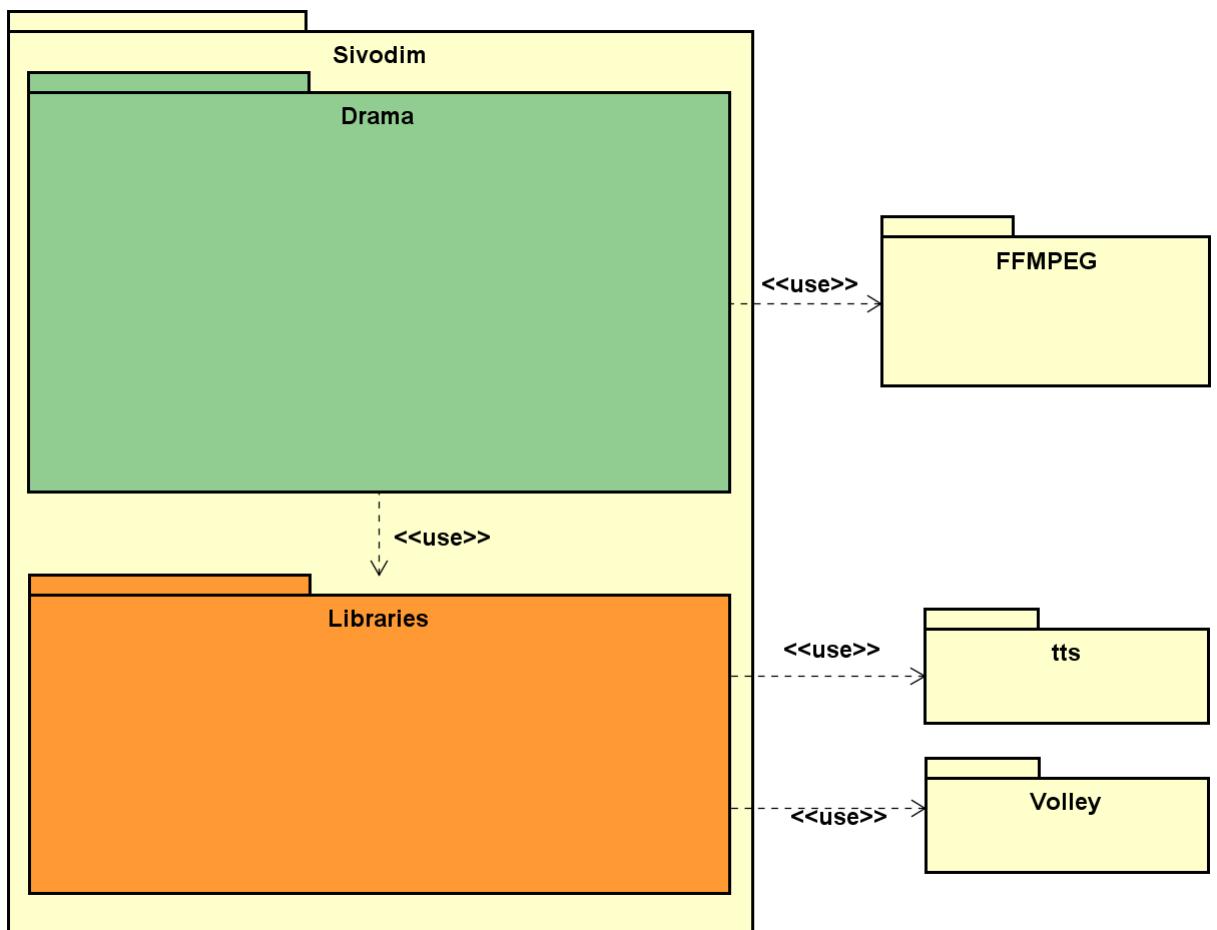


Figura 1: Componente Sivodim

3.2 Sivodim::Libraries

3.2.1 Informazioni sul package

Di seguito viene illustrata, tramite un diagramma delle classi, la visione di insieme del package **Libraries** contenente tutte le classi relative all'applicazione SiVoDiM (app dedicata alla configurazione delle voci attraverso FA-TTS_G).

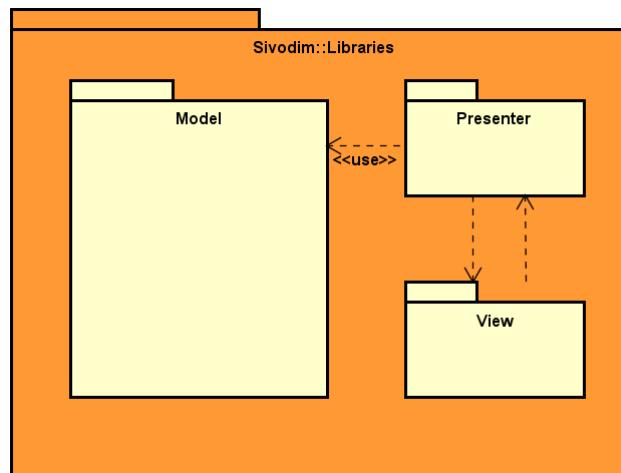


Figura 2: Componente Sivodim::Libraries

3.3 Sivodim::Libraries::Model

3.3.1 Informazioni sul package

Di seguito viene illustrata, tramite un diagramma delle classi, la visione di insieme del package Libraries::Model contenente tutte le classi relative al Model nel rispetto del pattern MVP.

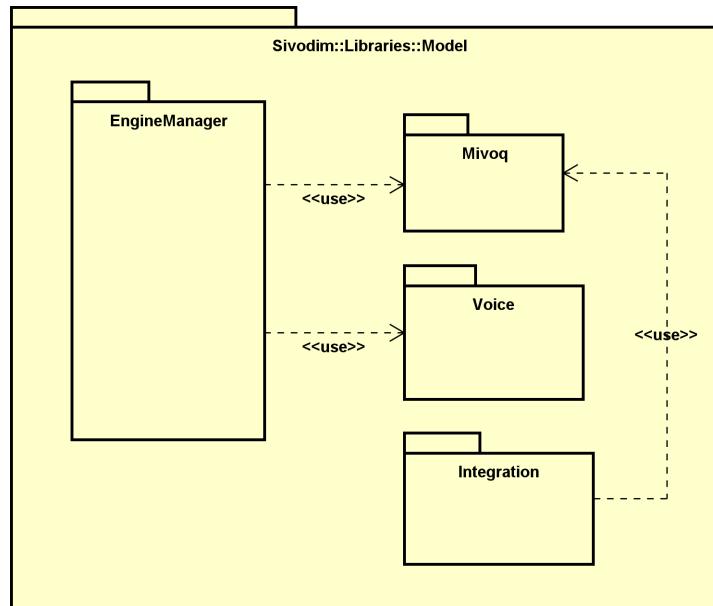


Figura 3: Componente Sivodim::Libraries::Model

3.4 Sivodim::Libraries::Model::EngineManager

3.4.1 Informazioni sul package

Di seguito viene illustrata, tramite un diagramma delle classi, la visione di insieme del package Libraries::Model::EngineManager contenente le classi utili alla gestione della selezione del

corretto motore di sintesi tra FA-TTS_G e quello di default di Android_G (nel caso di assenza di connessione di rete).

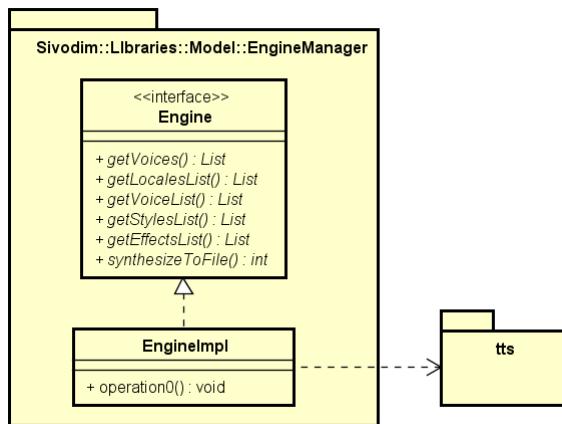


Figura 4: Componente Sivodim::Libraries::Model::EngineManager

3.4.2 Engine (interface)

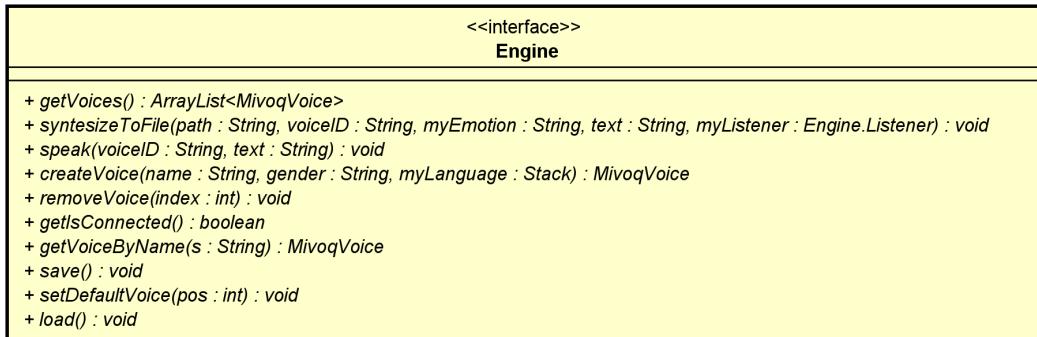


Figura 5: Classe Sivodim::Libraries::Model::EngineManager::Engine

3.4.2.1 Descrizione

Interfaccia che rappresenta il contratto attraverso cui implementare il TTS_G di sistema utilizzando se disponibile il servizio MIVOQ s.r.l..

3.4.2.2 Utilizzo

Viene usata per definire il contratto attraverso cui utilizzare i servizi di TTS.

3.4.2.3 Classi figlie

- `Sivodim::Libraries::Model::EngineManager::EngineImpl`.

3.4.2.4 Metodi

3.4.2.4.1 createVoice(String name, String gender, String myLanguage)

- **Accesso:** public;
- **Tipo di ritorno:** MivoqVoice;
- **Argomenti:**
 - name: Stringa contenente il nome della voce che si vuole creare;
 - gender: Stringa contenente il genere della voce che si vuole creare;
 - myLanguage: Stringa contenente la lingua della voce che si vuole costruire.
- **Descrizione:** Metodo della classe Engine che crea e ritorna un nuovo oggetto di tipo MivoqVoice.

3.4.2.4.2 getVoices()

- **Accesso:** public;
- **Tipo di ritorno:** ArrayList<MivoqVoice>;
- **Descrizione:** Metodo che ritorna la lista di voci personalizzate create dall'utente.

3.4.2.4.3 removeVoice(int index)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - index: indice della voce che si vuole andare a eliminare.
- **Descrizione:** Metodo per l'eliminazione delle voci create dall'utente non più necessarie.

3.4.2.4.4 speak(String voiceID, String text)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - voiceID: Stringa contenente l'ID della voce con cui sintetizzare il testo;
 - text: Stringa contenente il testo da sintetizzare.
- **Descrizione:** Metodo che permette di sintetizzare e riprodurre un testo con una voce desiderata tra quelle create dall'utente. La sintesi e la riproduzione dell'audio avvengono in un thread esterno da quello principale, per non bloccare rendere la richiesta bloccante.

3.4.2.4.5 `synthesizeToFile(String path, String voiceID, String myEmotion, String text, Engine.listener myListener)`

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - path: Stringa contenente l'indirizzo del file su cui salvare la sintesi vocale;
 - voiceID: Stringa contenente l'ID della voce con cui sintetizzare il testo;
 - myEmotion: Stringa contenente l'ID dell'emozione da applicare alla voce;
 - text: Stringa contenente il testo da sintetizzare;
 - myListener: Oggetto contenente il metodo da richiamare alla fine delle operazioni di sintesi vocale.
- **Descrizione:** Metodo che permette la sintesi di un testo dato su file. Il metodo permette di impostare un'emozione alla sintesi, e di definire le operazioni da fare una volta completata la sintesi. Il tutto viene gestito in un thread esterno da quello principale, per non bloccare l'esecuzione.

3.4.2.4.6 `getIsConnected()`

- **Accesso:** public;
- **Tipo di ritorno:** boolean;
- **Descrizione:** Metodo che permette di verificare se esiste una connessione internet attiva nel dispositivo.

3.4.2.4.7 `save()`

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Descrizione:** Metodo che permette di salvare tutte le voci nell'Engine.

3.4.2.4.8 `load()`

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Descrizione:** Metodo che permette di caricare tutte le voci nell'Engine.

3.4.2.4.9 getVoiceByName(String s)

- **Accesso:** public;
- **Tipo di ritorno:** MivoqVoice;
- **Argomenti:**
 - s: Stringa contenente il nome della voce da ritornare.
- **Descrizione:** Metodo che permette di reperire una voce nell'Engine.

3.4.2.4.10 setDefaultVoice(int pos)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - pos: intero corrispondente alla posizione della voce nell'array da salvare come voce di default.
- **Descrizione:** Metodo che permette di reperire una voce nell'Engine.

3.4.3 Engine.Listener (interface)

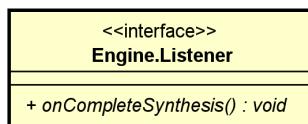


Figura 6: Classe Sivodim::Libraries::Model::EngineManager::Engine.Listener

3.4.3.1 Descrizione

Interfaccia per la creazione di classi anonime.

3.4.3.2 Utilizzo

Definisce classi anonime che contengono il comportamento da eseguire al termine della sintesi vocale.

3.4.3.3 Metodi

3.4.3.3.1 onCompleteSynthesis()

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Descrizione:** Metodo che a seconda della classe che lo implementa esegue delle operazioni specifiche in seguito alla sintesi vocale.

3.4.4 EngineImpl (class)

EngineImpl
<pre> - myEngine : MivoqTTSSingleton - backupEngine : TextToSpeech - myContext : Context - isConnected : boolean + getVoices() : ArrayList<MivoqVoice> + synthesizeToFile(path : String, voiceID : String, myEmotion : String, text : String, myListener : Engine.Listener) : void + speak(voiceID : String, text : String) : void + createVoice(name : String, gender : String, myLanguage : Stack) : MivoqVoice + removeVoice(index : int) : void + save() : void + load() + getVoiceByName(s : String) : MivoqVoice + getIsConnected() : boolean + setDefaultVoice(pos : int) : void + EngineImpl(c : Context) </pre>

Figura 7: Classe Sivodim::Libraries::Model::EngineManager::EngineImpl

3.4.4.1 Descrizione

Classe che implementa l'interfaccia Engine.

3.4.4.2 Utilizzo

Implementa il contratto della classe Engine che gestisce la sintesi vocale per la libreria Sivodim.

3.4.4.3 Classi ereditate

- Sivodim::Libraries::Model::EngineManager::Engine.

3.4.4.4 Attributi

3.4.4.4.1 isConnected

- **Accesso:** private;
- **Tipo:** boolean;
- **Descrizione:** Flag che indica la presenza o assenza di una connessione di rete.

3.4.4.4.2 myContext

- **Accesso:** private;
- **Tipo:** android.content.Context;
- **Descrizione:** Attributo in cui viene salvato il contesto di inizializzazione dell'Engine.

3.4.4.4.3 backupEngine

- **Accesso:** private;
- **Tipo:** android.speech.tts.TextToSpeech;
- **Descrizione:** Attributo statico in cui viene salvato il motore di sintesi alternativo in caso di mancata connessione Internet.

3.4.4.4 myEngine

- **Accesso:** private;
- **Tipo:** MivoqTTSSingleton;
- **Descrizione:** Attributo statico dell'Engine principale di sintesi vocale, tramite il servizio offerto dai server MIVOQ s.r.l.; il campo è statico in quanto riferimento ad una classe implementata con pattern Singleton;

3.4.4.5 Metodi

3.4.4.5.1 EnginelImpl(Context c)

- **Accesso:** public;
- **Argomenti:**
 - c: Contesto di inizializzazione del motore di sintesi.
- **Descrizione:** Costruttore della classe EnginelImpl.

3.4.4.5.2 createVoice(String name, String gender, String myLanguage)

- **Accesso:** public;
- **Tipo di ritorno:** MivoqVoice;
- **Argomenti:**
 - name: Stringa contenente il nome da assegnare alla voce personalizzata;
 - gender: Stringa contenente il genere della voce da creare;
 - myLanguage: Stringa contenente la lingua della voce da creare.
- **Descrizione:** Metodo per la creazione di una nuova voce personalizzata, in base ai valori passati come parametri del metodo, all'interno del motore di sintesi.

3.4.4.5.3 getVoices()

- **Accesso:** public;
- **Tipo di ritorno:** ArrayList<MivoqVoice>;
- **Descrizione:** Metodo che ritorna la lista delle voci personalizzate create dall'utente.

3.4.4.5.4 removeVoice(int index)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - index: indice della voce che si vuole rimuovere dall'engine.
- **Descrizione:** Metodo per la rimozione di una voce dal motore di sintesi.

3.4.4.5.5 speak(String voiceID, String text)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - voiceID: Stringa contenente l'ID della voce con cui si vuole sintetizzare il testo;
 - text: Stringa contenente il testo da sintetizzare
- **Descrizione:** Implementazione del metodo dell'interfaccia che permette di riprodurre una stringa di testo con una determinata voce.

3.4.4.5.6 synthesizeToFile(String path, String voiceID, String myEmotion, String text, Engine.listener myListener)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - path: Stringa contenente l'indirizzo in cui andare a salvare il file audio sintetizzato;
 - voiceID: Stringa contenente l'ID della voce con cui sintetizzare il testo;
 - myEmotion: Stringa contenente l'ID dell'emozione da applicare alla voce con cui sintetizzare il testo;
 - text: Stringa contenente il testo da sintetizzare;
 - myListener: Oggetto contenente il metodo da invocare quando la sintesi vocale su file si è conclusa.
- **Descrizione:** Implementazione del metodo dell'interfaccia che permette la sintesi vocale su file di un testo, con una determinata voce ed una emozione.

3.4.4.5.7 getIsConnected()

- **Accesso:** public;
- **Tipo di ritorno:** boolean;
- **Descrizione:** Metodo che permette di verificare se esiste una connessione internet attiva nel dispositivo.

3.4.4.5.8 save()

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Descrizione:** Metodo che permette di salvare tutte le voci create dall'utente in un file xml.

3.4.4.5.9 load()

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Descrizione:** Metodo che permette di caricare tutte le voci nell'Engine dal file xml.

3.4.4.5.10 getVoiceByName(String s)

- **Accesso:** public;
- **Tipo di ritorno:** MivoqVoice;
- **Argomenti:**
 - s: Stringa contenente il nome della voce da ritornare.
- **Descrizione:** Metodo che permette di reperire una voce nell'Engine.

3.4.4.5.11 setDefaultVoice(int pos)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - pos: intero corrispondente alla posizione della voce nell'array da salvare come voce di default.
- **Descrizione:** Metodo che permette di reperire una voce nell'Engine.

3.5 Sivodim::Libraries::Model::Integration

3.5.1 Informazioni sul package

Di seguito viene illustrata, tramite un diagramma delle classi, la visione di insieme del package Libraries::Model::Integration contenente tutte le classi usate dal sistema operativo per visualizzare alcune feature all'interno del menù di impostazione di Android_G nel momento in cui ci sia la selezione del motore FA-TTS_G come predefinito.

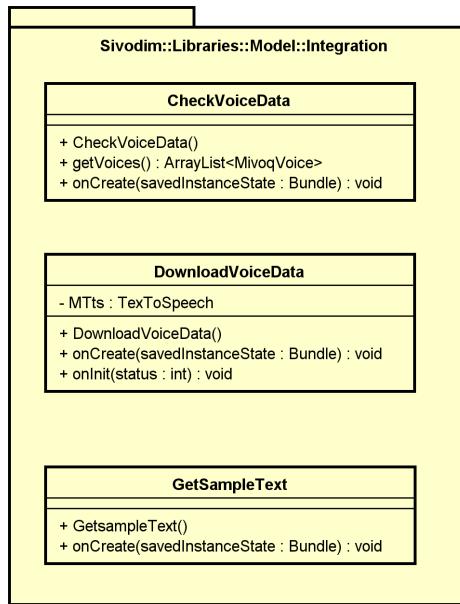


Figura 8: Componente Sivodim::Libraries::Model::Integration

3.5.2 CheckVoiceData (class)

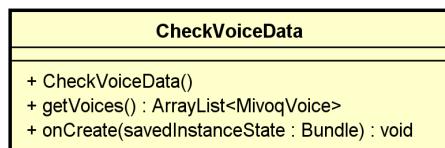


Figura 9: Classe Sivodim::Libraries::Model::Integration::CheckVoiceData

3.5.2.1 Descrizione

Classe che comunica al sistema tutte le voci disponibili create dall'utente.

3.5.2.2 Utilizzo

Viene utilizzata per rendere disponibili al sistema le voci create dall'utente con l'applicazione di configurazione.

3.5.2.3 Metodi

3.5.2.3.1 CheckVoiceData()

- **Accesso:** public;
- **Descrizione:** Costruttore della classe CheckVoiceData.

3.5.2.3.2 getVoices()

- **Accesso:** public;
- **Tipo di ritorno:** ArrayList<MivoqVoice>;
- **Descrizione:** Metodo reperisce le voci dell'Engine e le rende visualizzabili nel menù di impostazione di sistema Android_G.

3.5.2.3.3 onCreate(Bundle savedInstanceState)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - savedInstanceState: parametro di sistema per creare l'activity. MIVOQ s.r.l.
- **Descrizione:** Metodo che crea l'activity e passare i dati attraverso l'intent.

3.5.3 DownloadVoiceData (class)

DownloadVoiceData
- MTts : TexToSpeech
+ DownloadVoiceData()
+ onCreate(savedInstanceState : Bundle) : void
+ onInit(status : int) : void

Figura 10: Classe Sivodim::Libraries::Model::Integration::DownloadVoiceData

3.5.3.1 Descrizione

Classe per il funzionamento del servizio di sintesi vocale del sistema.

3.5.3.2 Utilizzo

Viene utilizzata per il download di dati necessari al servizio di sintesi vocali.

3.5.3.3 Attributi

3.5.3.3.1 MTts

- **Accesso:** private;
- **Tipo:** TextToSpeech;
- **Descrizione:** Riferimento a un'istanza della classe di sintesi vocale di sistema.

3.5.3.4 Metodi

3.5.3.4.1 DownloadVoiceData()

- **Accesso:** public;
- **Descrizione:** Costruttore della classe DownloadVoiceData.

3.5.3.4.2 onInit(int status)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Descrizione:** Metodo che inizializza il TextToSpeech.

3.5.3.4.3 onCreate(Bundle savedInstanceState)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - savedInstanceState: parametro di sistema per creare l'activity. MIVOQ s.r.l.
- **Descrizione:** Metodo che crea l'activity e passare i dati attraverso l'intent.

3.5.4 GetSampleText (class)

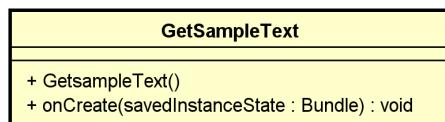


Figura 11: Classe Sivodim::Libraries::Model::Integration::GetsampleText

3.5.4.1 Descrizione

Classe che si occupa di trasmettere al sistema il testo di esempio da leggere per le varie lingue.

3.5.4.2 Utilizzo

Viene utilizzata per trasmettere al sistema il testo di esempio per testare la voce di sistema selezionata.

3.5.4.3 Metodi

3.5.4.3.1 GetSampleText()

- **Accesso:** public;
- **Descrizione:** Costruttore della classe GetSampleText.

3.5.4.3.2 onCreate(Bundle savedInstanceState)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - savedInstanceState: parametro di sistema per creare l'activity. MIVOQ s.r.l.
- **Descrizione:** Metodo che crea l'activity e passare i dati attraverso l'intent.

3.6 Sivodim::Libraries::Model::Mivoq

3.6.1 Informazioni sul package

Di seguito viene illustrata, tramite un diagramma delle classi, la visione di insieme del package Libraries::Model::Mivoq contenente tutte le classi che si occupano di gestire la richiesta di sintesi vocale verso MIVOQ, del salvataggio delle voci nella memoria del dispositivo e dell'integrazione del motore FA-TTS_G con il sistema operativo Android_G.

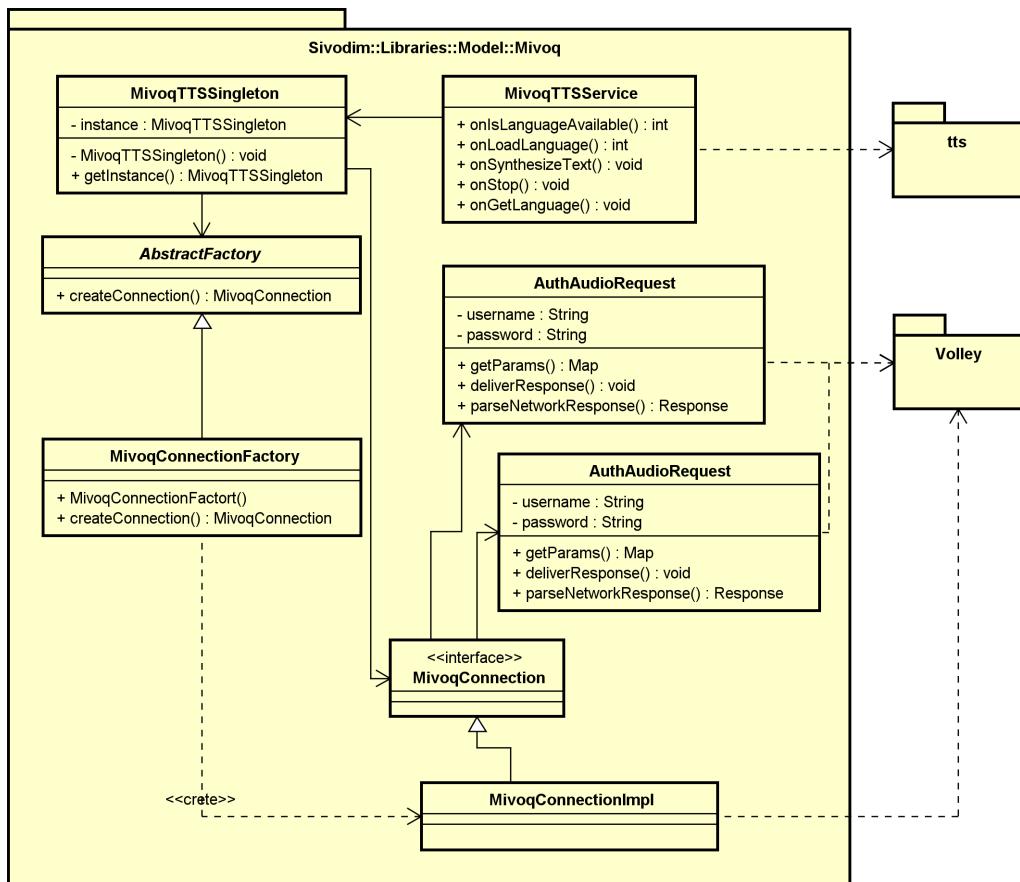


Figura 12: Componente Sivodim::Libraries::Model::Mivoq

3.6.2 AbstractFactory (abstract)

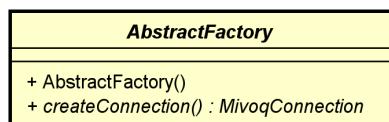


Figura 13: Classe Sivodim::Libraries::Model::Mivoq::AbstractFactory

3.6.2.1 Descrizione

Classe astratta che rappresenta il contratto della classe Factory che ha il compito di creare istanze di oggetti di tipo **MivoqConnection**.

3.6.2.2 Utilizzo

Viene utilizzata per creare istanze di oggetti di tipo **MivoqConnection**.

3.6.2.3 Classi figlie

- Sivodim::Libraries::Model::Mivoq::MivoqConnectionFactory.

3.6.2.4 Metodi

3.6.2.4.1 createConnection()

- **Accesso:** public;
- **Tipo di ritorno:** MivoqConnection;
- **Descrizione:** Metodo astratto per la creazione di un oggetto che implementa l'interfaccia MivoqConnection.

3.6.3 MivoqConnectionFactory (class)

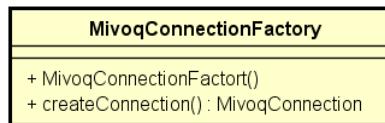


Figura 14: Classe Sivodim::Libraries::Model::Mivoq::MivoqConnectionFactory

3.6.3.1 Descrizione

Classe che implementa la classe AbstractFactory per la creazione di oggetti di tipo MivoqConnectionImpl.

3.6.3.2 Utilizzo

Viene utilizzato per istanziare oggetti concreti di tipo MivoqConnectionImpl.

3.6.3.3 Classi ereditate

- Sivodim::Libraries::Model::Mivoq::AbstractFactory.

3.6.3.4 Metodi

3.6.3.4.1 MivoqConnectionFactory()

- **Accesso:** public;
- **Descrizione:** Costruttore della classe MivoqConnectionFactory.

3.6.3.4.2 createConnection()

- **Accesso:** public;
- **Tipo di ritorno:** MivoqConnection;
- **Descrizione:** Implementazione del metodo dell'interfaccia, che costruisce un oggetto MivoqConnectionImpl.

3.6.4 MivoqConnection (interface)

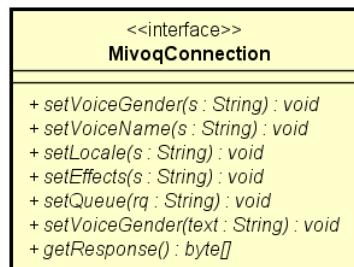


Figura 15: Classe Sivodim::Libraries::Model::Mivoq::MivoqConnection

3.6.4.1 Descrizione

Interfaccia che rappresenta il contratto attraverso cui implementare la connessione ai server MIVOQ s.r.l..

3.6.4.2 Utilizzo

Viene usata per definire il contratto di interazione tra l' $engine_G$ e il server.

3.6.4.3 Classi figlie

- Sivodim::Libraries::Model::Mivoq::MivoqConnectionImpl.

3.6.4.4 Metodi

3.6.4.4.1 getResponse()

- **Accesso:** public;
- **Tipo di ritorno:** byte[];
- **Descrizione:** Metodo che ritorna la risposta alla richiesta http, se non è ancora disponibile ritorna invece null.

3.6.4.4.2 sendRequest(String text)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - text: Stringa che contiene il testo che deve essere sintetizzato dai server MIVOQ s.r.l.
- **Descrizione:** Metodo per l'invio della richiesta ai server MIVOQ s.r.l..

3.6.4.4.3 setEffects(String s)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - s: Stringa contenente la codifica degli effetti da applicare alla sintetizzazione del testo.
- **Descrizione:** Metodo per settare gli effetti da applicare alla sintesi vocale.

3.6.4.4.4 setLocale(String s)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - s: Stringa contenente la codifica della lingua in cui sintetizzare il testo.
- **Descrizione:** Metodo per l'impostazione della lingua in cui sintetizzare il testo.

3.6.4.4.5 setQueue(RequestQueue rq)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - rq: Oggetto condiviso che gestisce l'inoltro delle richieste http da parte del sistema.
- **Descrizione:** Metodo per l'impostazione della coda di inoltro delle richieste http.

3.6.4.4.6 setVoiceGender(String s)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - s: Stringa contenente il genere della voce con cui sintetizzare il testo.
- **Descrizione:** Metodo per l'impostazione del genere della voce da utilizzare per la sintesi del testo.

3.6.4.4.7 setVoiceName(String s)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - s: Stringa contenente il nome della voce da utilizzare nei server MIVOQ s.r.l.
- **Descrizione:** Metodo per l'impostazione della voce di sistema da utilizzare per la sintesi vocale sui server MIVOQ s.r.l..

3.6.5 MivoqConnectionImpl (class)

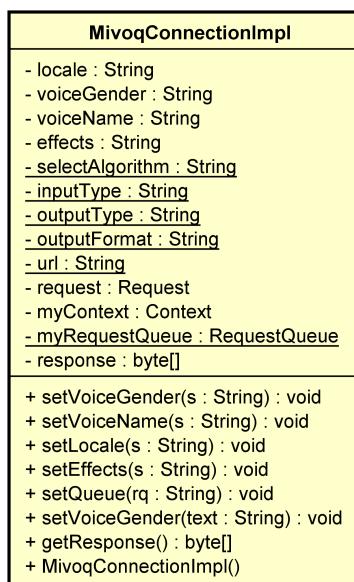


Figura 16: Classe Sivodim::Libraries::Model::Mivoq::MivoqConnectionImpl

3.6.5.1 Descrizione

Classe che implementa l'interfaccia MivoqConnection.

3.6.5.2 Utilizzo

Implementa il contratto di interazione con il protocollo attuale di connessione al server di MIVOQ s.r.l..

3.6.5.3 Classi ereditate

- Sivodim::Libraries::Model::MivoqConnection.

3.6.5.4 Attributi

3.6.5.4.1 effects

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Attributo stringa in cui viene memorizzato l'insieme degli effetti codificati per la richiesta.

3.6.5.4.2 inputType

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Attributo statico costante che contiene il tipo di input memorizzato.

3.6.5.4.3 locale

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Attributo codificato che contiene la lingua in cui deve essere fatta la sintesi.

3.6.5.4.4 myContext

- **Accesso:** private;
- **Tipo:** Context;
- **Descrizione:** Attributo del contesto di inizializzazione per le richieste.

3.6.5.4.5 myRequestQueue

- **Accesso:** private;
- **Tipo:** com.android.volley.RequestQueue;
- **Descrizione:** Attributo per la gestione della coda di richieste ai server MIVOQ s.r.l..

3.6.5.4.6 outputFormat

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Attributo statico costante che memorizza il formato di ricezione della richiesta.

3.6.5.4.7 outputType

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Attributo statico costante che memorizza il tipo di output della richiesta.

3.6.5.4.8 request

- **Accesso:** private;
- **Tipo:** com.android.volley.Request;
- **Descrizione:** Attributo che memorizza l'oggetto che gestisce la richiesta ai server di MIVOQ s.r.l..

3.6.5.4.9 response

- **Accesso:** private;
- **Tipo:** byte[];
- **Descrizione:** Attributo in cui è memorizzata la risposta alla richiesta http_G.

3.6.5.4.10 selectAlgorithm

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Attributo statico costante utilizzato per selezionare l'algoritmo utilizzato per la sintesi vocale.

3.6.5.4.11 url

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Attributo statico costante in cui è memorizzato l'indirizzo del servizio offerto da MIVOQ s.r.l. per la sintesi vocale.

3.6.5.4.12 voiceGener

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Attributo in cui è memorizzato il genere della voce.

3.6.5.4.13 voiceName

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Attributo in cui è memorizzato il nome della voce da utilizzare per la sintesi del testo.

3.6.5.5 Metodi

3.6.5.5.1 getResponse()

- **Accesso:** public;
- **Tipo di ritorno:** byte[];
- **Descrizione:** Implementazione del metodo dell'interfaccia, che permette l'accesso alla risposta della richiesta http.

3.6.5.5.2 sendRequest(String text)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - text: Testo di cui richiedere la sintesi.
- **Descrizione:** Implementazione del metodo dell'interfaccia per l'invio della richiesta di sintesi ai server MIVOQ s.r.l.. Il metodo istanzia un oggetto di tipo AudioRequest, che si occupa dell'invio della richiesta ai server, e la mette in una coda di richieste comune.

3.6.5.5.3 setEffects(String s)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - s: Stringa contenente la codifica degli effetti da applicare alla sintesi vocale.
- **Descrizione:** Implementazione del metodo dell'interfaccia, per l'impostazione degli effetti da applicare alla richiesta di sintesi.

3.6.5.5.4 setLocale(String s)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - s: Stringa contenente la codifica della lingua con cui fare la sintesi vocale.
- **Descrizione:** Implementazione del metodo dell'interfaccia, per l'impostazione della lingua con cui fare la sintesi vocale.

3.6.5.5.5 setQueue(RequestQueue rq)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - rq: Parametro per l'impostazione della coda condivisa delle richieste.
- **Descrizione:** Implementazione del metodo dell'interfaccia, che permette l'impostazione della coda per le richieste.

3.6.5.5.6 setVoiceGender(String s)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - s: Stringa contenente la codifica del genere della voce con cui sintetizzare il testo.
- **Descrizione:** Implementazione del metodo dell'interfaccia, che permette di impostare il genere della voce con cui sintetizzare il testo.

3.6.5.5.7 setVoiceName(String s)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - s: Stringa contenente il nome della voce di sistema con cui sintetizzare il testo.
- **Descrizione:** Implementazione del metodo dell'interfaccia, che permette l'impostazione della voce di sistema con cui sintetizzare il testo.

3.6.5.5.8 MivoqConnectionImpl()

- **Accesso:** public;
- **Descrizione:** Costruttore della classe MivoqConnectionImpl.

3.6.6 MivoqTTSSingleton (class)

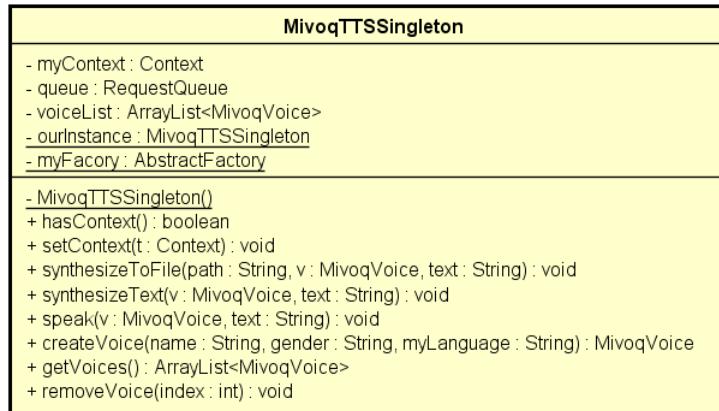


Figura 17: Classe Sivodim::Libraries::Model::Mivoq::MivoqTTSSingleton

3.6.6.1 Descrizione

Classe Singleton_G che implementa l'*engine_G* di sintesi vocale.

3.6.6.2 Utilizzo

Viene utilizzato per la sintesi vocale attraverso i server MIVOQ s.r.l..

3.6.6.3 Attributi

3.6.6.3.1 myContext

- **Accesso:** private;
- **Tipo:** Context;
- **Descrizione:** Attributo di memorizzazione del contesto di inizializzazione dell'oggetto.

3.6.6.3.2 myFactory

- **Accesso:** private;
- **Tipo:** AbstractFactory;
- **Descrizione:** Attributo statico che gestisce la creazione delle connessioni verso i server MIVOQ s.r.l..

3.6.6.3.3 ourInstance

- **Accesso:** private;
- **Tipo:** MivoqTTSSingleton;
- **Descrizione:** Attributo statico che permette la realizzazione del *Design Pattern_G Singleton_G*.

3.6.6.3.4 queue

- **Accesso:** private;
- **Tipo:** com.android.volley.RequestQueue;
- **Descrizione:** Attributo per la gestione della lista di richieste verso il server MIVOQ s.r.l..

3.6.6.3.5 voiceList

- **Accesso:** private;
- **Tipo:** java.util.ArrayList<MivoqVoice>;
- **Descrizione:** Attributo dove vengono memorizzate le voci create dall'utente.

3.6.6.4 Metodi

3.6.6.4.1 MivoqTTSSingleton()

- **Accesso:** private;
- **Descrizione:** Costruttore privato della classe MivoqTTSSingleton.

3.6.6.4.2 createVoice(String name, String gender, String myLanguage)

- **Accesso:** public;
- **Tipo di ritorno:** MivoqVoice;
- **Argomenti:**
 - name: Nome da assegnare alla nuova voce che si sta creando;
 - gender: Genere della lingua della nuova voce che si sta creando;
 - myLanguage: Lingua della nuova voce che si sta creando.
- **Descrizione:** Metodo per la creazione di una nuova voce personalizzata per l'utente.

3.6.6.4.3 getInstance()

- **Accesso:** public;
- **Tipo di ritorno:** MivoqTTSSingleton;
- **Descrizione:** Metodo statico per l'acquisizione dell'istanza del MivoqTTSSingleton.

3.6.6.4.4 getVoices()

- **Accesso:** public;
- **Tipo di ritorno:** ArrayList<MivoqVoice>;
- **Descrizione:** Metodo per l'acquisizione delle voci personali create dall'utente, utilizzabili per la sintesi vocale.

3.6.6.4.5 hasContext()

- **Accesso:** public;
- **Tipo di ritorno:** boolean;
- **Descrizione:** Metodo per la verifica dell'assegnazione di un contesto al Singleton.

3.6.6.4.6 setContext(Context T)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - T: Contesto da assegnare al singleton per la creazione delle richieste http e la gestione dei file.
- **Descrizione:** Metodo per l'assegnazione di un contesto alla classe MivoqTTSSingleton.

3.6.6.4.7 removeVoice(int index)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - index: Parametro per identificare la posizione della voce da eliminare.
- **Descrizione:** Metodo per la rimozione di voci dalla lista di voci personali create dall'utente.

3.6.6.4.8 speak(MivoqVoice v, String text)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - v: Voce creata dall'utente con cui sintetizzare il testo;
 - text: Stringa di testo che deve essere sintetizzato.
- **Descrizione:** Metodo per la sintetizzazione e riproduzione di una stringa di testo con una voce specifica creata dall'utente.

3.6.6.4.9 synthesizeToFile(String path, MivoqVoice v, String text)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - path: Stringa che indica dove salvare il file che è stato sintetizzato;
 - v: Voce personalizzata dell'utente, con cui il testo deve essere sintetizzato;
 - text: Testo da sintetizzare su file.
- **Descrizione:** Metodo per la sintesi vocale di testo su uno specifico file.

3.6.6.4.10 synthesizeText(MivoqVoice v, String text)

- **Accesso:** public;
- **Tipo di ritorno:** byte[];
- **Argomenti:**
 - v: Voce personalizzata dall'utente con cui sintetizzare il testo;
 - text: Testo che deve essere sintetizzato.
- **Descrizione:** Metodo che ritorna un array di byte contenente la sintesi vocale del testo in formato WAV_G.

3.6.7 MivoqTTSService (class)

MivoqTTSService	
-	mCallback : SynthesisCallback
+	MivoqTTSService()
#	onIsLanguageAvailable(lang : String, country : String, variant : String) : int
#	onGetLanguage() : String[]
#	onLoadLanguage(lang : String, country : String, variant : String) : int
#	onStop() : void
#	onSynthesisText(req : SynthesisRequest, call : SynthesisCallback) : void

Figura 18: Classe Sivodim::Libraries::Model::Mivoq::MivoqTTSService

3.6.7.1 Descrizione

Classe concreta che implementa il contratto di TextToSpeechService, per rendere disponibile al sistema il servizio di sintesi vocale di MIVOQ s.r.l.. La classe definisce i suoi metodi con visibilità protected poiché ridefiniscono i metodi di una classe di sistema.

3.6.7.2 Utilizzo

Viene utilizzata per rendere disponibile al sistema Android_G il servizio di sintesi vocale di MIVOQ s.r.l..

3.6.7.3 Classi ereditate

- tts::TextToSpeechService.

3.6.7.4 Attributi

3.6.7.4.1 mCallback

- **Accesso:** private;
- **Tipo:** android.speech.tts.SynthesisCallback;
- **Descrizione:** Attributo per la gestione dell'avvenuta sintesi per il sistema di sintesi di Android.

3.6.7.5 Metodi

3.6.7.5.1 MivoqTTSService()

- **Accesso:** public;
- **Descrizione:** Costruttore della classe MivoqTTSService.

3.6.7.5.2 onGetLanguage()

- **Accesso:** protected;
- **Tipo di ritorno:** String[];
- **Descrizione:** Metodo che ritorna la lingua, lo stato e la variante della lingua che è attualmente in uso all'interno dell'engine.

3.6.7.5.3 onIsLanguageAvailable(String lang, String country, String variant)

- **Accesso:** protected;
- **Tipo di ritorno:** int;
- **Argomenti:**
 - lang: Stringa contenente la lingua di cui si richiede la disponibilità;
 - country: Stringa contenente lo stato della lingua di cui si richiede la disponibilità;
 - variant: Stringa contenente la variante della lingua di cui si richiede la disponibilità.
- **Descrizione:** Metodo che controlla se il motore di sintesi vocale supporta la sintesi vocale in una determinata lingua, con variante statale o di zona.

3.6.7.5.4 onLoadLanguage(String lang, String country, String variant)

- **Accesso:** protected;
- **Tipo di ritorno:** int;
- **Argomenti:**
 - lang: Stringa contenente la lingua di cui si richiede la disponibilità;
 - country: Stringa contenente lo stato della lingua di cui si richiede la disponibilità;
 - variant: Stringa contenente la variante della lingua di cui si richiede la disponibilità.
- **Descrizione:** Metodo che segnala al motore di sintesi vocale di predisporsi per sintetizzare il testo nella lingua specificata.

3.6.7.5.5 onStop()

- **Accesso:** protected;
- **Tipo di ritorno:** void;
- **Descrizione:** Metodo che notifica al motore di sintesi vocale di fermare qualunque sintesi vocale che sta eseguendo.

3.6.7.5.6 onSynthesizeText(SynthesisRequest req, SynthesisCallback call)

- **Accesso:** protected;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - req: Oggetto contenente tutti i parametri con cui fare la sintesi vocale;
 - call: oggetto contenente la risposta alla richiesta di sintesi vocale;
- **Descrizione:** Metodo per eseguire la sintesi vocale sul testo dato.

3.6.8 AudioRequest (class)

AudioRequest
<pre> - mErrorListener : ErrorListener - mListener : Listener<byte[]> - delivered : boolean - mParams : Map<String, String> - responseHeaders : Map<String, String> + AudioRequest(post : int, mUrl : String, params : Map<String, String>, listener : Listener<byte[]>, errorListener : ErrorListener) + getHeaders() : Map<String, String> # getParams() : Map<String, String> # deliverResponse(response : byte[]) : void # parseNetworkError(volleyError : VolleyError) : VolleyError </pre>

Figura 19: Classe Sivodim::Libraries::Model::Mivoq::AudioRequest

3.6.8.1 Descrizione

Estensione della classe Request del pacchetto Volley.

3.6.8.2 Utilizzo

Viene utilizzata per gestire una risposta in formato audio da una richiesta http_G generica.

3.6.8.3 Attributi

3.6.8.3.1 responseHeaders

- **Accesso:** private;
- **Tipo:** java.util.Map<java.lang.String,java.lang.String>;
- **Descrizione:** Attributo per la libreria Volley in cui vengono salvati gli *headers* della risposta alla richiesta http_G .

3.6.8.3.2 mParams

- **Accesso:** private;
- **Tipo:** java.util.Map<java.lang.String,java.lang.String>;
- **Descrizione:** Attributo in cui vengono memorizzati i parametri da passare nella richiesta con il metodo POST.

3.6.8.3.3 delivered

- **Accesso:** private;
- **Tipo:** boolean;
- **Descrizione:** Attributo di controllo che verifica la ricezione della risposta.

3.6.8.3.4 mListener

- **Accesso:** private;
- **Tipo:** com.android.volley.Response.Listener<byte[]>;
- **Descrizione:** Attributo che memorizza una classe contenitore in cui è implementato il metodo per la gestione della risposta.

3.6.8.3.5 mErrorListener

- **Accesso:** private;
- **Tipo:** com.android.volley.Response.ErrorListener;
- **Descrizione:** Attributo che memorizza una classe contenitore in cui è implementato il metodo per la gestione di un errore nella risposta.

3.6.8.4 Metodi

3.6.8.4.1 AudioRequest(int post, String mUrl, Map<String, String> params, Response.Listener<byte[]> listener, Response.ErrorListener errorListener)

- **Accesso:** public;
- **Argomenti:**
 - post: Intero che indica il metodo di invio dei parametri (POST,GET);
 - mUrl: Indirizzo di invio della richiesta;
 - params: Mappa contenente i parametri da passare alla richiesta;
 - listener: Oggetto contenente il metodo da invocare all'arrivo della risposta;
 - errorListener: Oggetto contenente il metodo da invocare in caso di errori nell'arrivo della risposta.
- **Descrizione:** Costruttore della AudioRequest.

3.6.8.4.2 deliverResponse(byte[] response)

- **Accesso:** protected;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - response: Vettore di Byte contenente l'audio della risposta alla richiesta.
- **Descrizione:** Metodo per la consegna della risposta ricevuta dalla richiesta http.

3.6.8.4.3 getHeaders()

- **Accesso:** public;
- **Tipo di ritorno:** Map<String, String>;
- **Descrizione:** Metodo che ritorna l'intestazione della richiesta http da inviare.

3.6.8.4.4 getParams()

- **Accesso:** protected;
- **Tipo di ritorno:** Map<String, String>;
- **Descrizione:** Metodo che ritorna i parametri da inviare nella richiesta http.

3.6.8.4.5 parseNetworkResponse(NetworkResponse response)

- **Accesso:** protected;
- **Tipo di ritorno:** Response<byte[]>;
- **Argomenti:**
 - response: oggetto contenente la risposta della rete alla richiesta.
- **Descrizione:** Metodo per l'elaborazione della risposta ricevuta dalla rete alla richiesta http.

3.6.9 AuthAudioRequest (class)

AuthAudioRequest
<pre> - mErrorListener : ErrorListener - mListener : Listener<byte> - delivered : boolean - mParams : Map<String, String> - responseHeaders : Map<String, String> - password : String - username : String + AuthAudioRequest(post : int, mUrl : String, params : Map<String, String>, listener : Listener<byte>, errorListener : ErrorListener) + getHeaders() : Map<String, String> # getParams() : Map<String, String> # deliverResponse(response : byte[]) : void # parseNetworkError(volleyError : VolleyError) : VolleyError </pre>

Figura 20: Classe Sivodim::Libraries::Model::Mivoq::AuthAudioRequest

3.6.9.1 Descrizione

Estensione della classe Request del pacchetto Volley.

3.6.9.2 Utilizzo

Viene utilizzata per gestire le richieste audio e l'autenticazione presso il server di MIVOQ s.r.l..

3.6.9.3 Attributi

3.6.9.3.1 username

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Attributo corrispondente all'username per effettuare la login.

3.6.9.3.2 password

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Attributo corrispondente alla password per effettuare la login.

3.6.9.3.3 responseHeaders

- **Accesso:** private;
- **Tipo:** java.util.Map<java.lang.String,java.lang.String>;
- **Descrizione:** Attributo per la libreria Volley in cui vengono salvati gli *headers* della risposta alla richiesta http_G .

3.6.9.3.4 mParams

- **Accesso:** private;
- **Tipo:** java.util.Map<java.lang.String,java.lang.String>;
- **Descrizione:** Attributo in cui vengono memorizzati i parametri da passare nella richiesta con il metodo POST.

3.6.9.3.5 delivered

- **Accesso:** private;
- **Tipo:** boolean;
- **Descrizione:** Attributo di controllo che verifica la ricezione della risposta.

3.6.9.3.6 mListener

- **Accesso:** private;
- **Tipo:** com.android.volley.Response.Listener<byte[]>;
- **Descrizione:** Attributo che memorizza una classe contenitore in cui è implementato il metodo per la gestione della risposta.

3.6.9.3.7 mErrorListener

- **Accesso:** private;
- **Tipo:** com.android.volley.Response.ErrorListener;
- **Descrizione:** Attributo che memorizza una classe contenitore in cui è implementato il metodo per la gestione di un errore nella risposta.

3.6.9.4 Metodi

3.6.9.4.1 AuthAudioRequest(int post, String mUrl, Map<String, String> params, Response.Listener<byte[]> listener, Response.ErrorListener errorListener)

- **Accesso:** public;
- **Argomenti:**
 - post: Intero che indica il metodo di invio dei parametri (POST,GET);
 - mUrl: Indirizzo di invio della richiesta;
 - params: Mappa contenente i parametri da passare alla richiesta;
 - listener: Oggetto contenente il metodo da invocare all'arrivo della risposta;
 - errorListener: Oggetto contenente il metodo da invocare in caso di errori nell'arrivo della risposta.
- **Descrizione:** Costruttore della AuthAudioRequest.

3.6.9.4.2 deliverResponse(byte[] response)

- **Accesso:** protected;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - response: Vettore di Byte contenente l'audio della risposta alla richiesta.
- **Descrizione:** Metodo per la consegna della risposta ricevuta dalla richiesta http.

3.6.9.4.3 getHeaders()

- **Accesso:** public;
- **Tipo di ritorno:** Map<String, String>;
- **Descrizione:** Metodo che ritorna l'intestazione della richiesta http da inviare.

3.6.9.4.4 getParams()

- **Accesso:** protected;
- **Tipo di ritorno:** Map<String, String>;
- **Descrizione:** Metodo che ritorna i parametri da inviare nella richiesta http.

3.6.9.4.5 parseNetworkResponse(NetworkResponse response)

- **Accesso:** protected;
- **Tipo di ritorno:** Response<byte[]>;
- **Argomenti:**
 - response: oggetto contenente la risposta della rete alla richiesta.
- **Descrizione:** Metodo per l'elaborazione della risposta ricevuta dalla rete alla richiesta http.

3.7 Sivodim::Libraries::Model::Voice

3.7.1 Informazioni sul package

Di seguito viene illustrata, tramite un diagramma delle classi, la visione di insieme del package Libraries::Model::Voice contenente tutte le classi che si occupano di gestire le voci e i loro effetti.

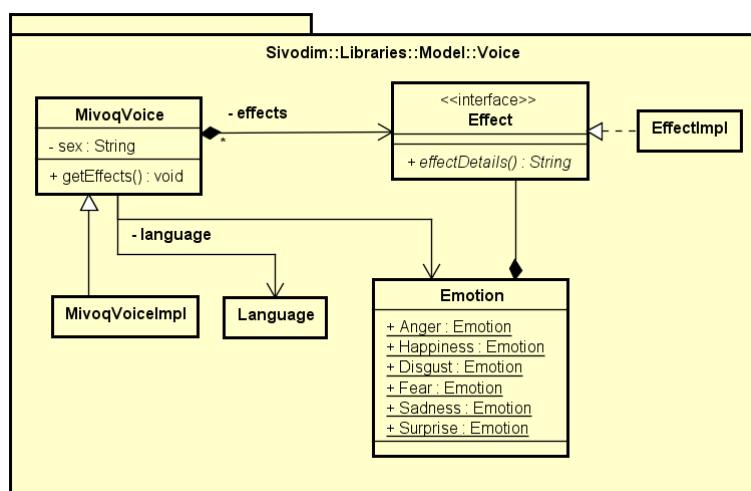


Figura 21: Componente Sivodim::Libraries::Model::Voice

3.7.2 MivoqVoice (class)

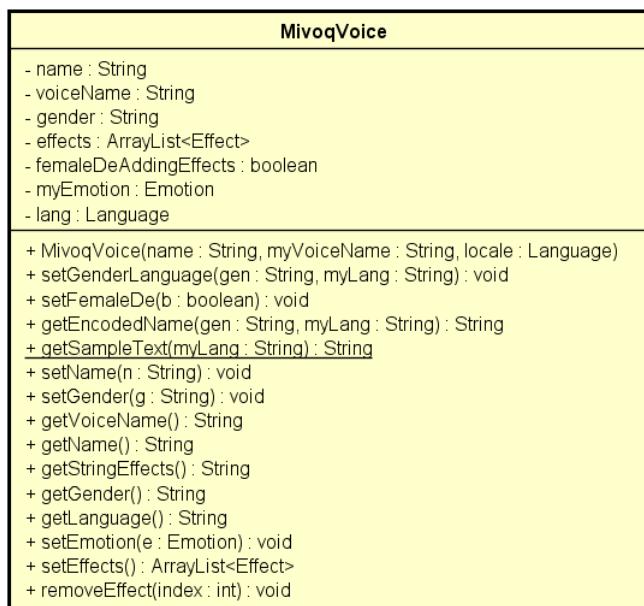


Figura 22: Classe Sivodim::Libraries::Model::Voice::MivoqVoice

3.7.2.1 Descrizione

Classe astratta che rappresenta una voce. È composta da una lingua ed un insieme di effetti ed emozioni.

3.7.2.2 Utilizzo

Viene definito il contratto di rappresentazione di una voce all'interno dell'*engine_G*.

3.7.2.3 Attributi

3.7.2.3.1 effects

- **Accesso:** private;
- **Tipo:** java.util.ArrayList<Effect>;
- **Descrizione:** Vettore in cui sono memorizzati gli effetti da applicare alla voce.

3.7.2.3.2 femaleDeAddingEffects

- **Accesso:** private;
- **Tipo:** boolean;
- **Descrizione:** Attributo per permettere l'utilizzo di voci femminili di tipo tedesco.

3.7.2.3.3 gender

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Attributo per la memorizzazione del genere della voce.

3.7.2.3.4 lang

- **Accesso:** private;
- **Tipo:** Language;
- **Descrizione:** Attributo per la memorizzazione della lingua della voce.

3.7.2.3.5 myEmotion

- **Accesso:** private;
- **Tipo:** Emotion;
- **Descrizione:** Attributo per l'applicazione di un'emozione alla voce.

3.7.2.3.6 name

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Attributo per la memorizzazione del nome associato dall'utente alla voce.

3.7.2.3.7 voiceName

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Attributo per la memorizzazione della voce di sistema da applicare.

3.7.2.4 Metodi

3.7.2.4.1 MivoqVoice(String name, String myVoiceName, Language locale)

- **Accesso:** public;
- **Argomenti:**
 - name: Stringa contenente un nome associato dall'utente alla voce;
 - myVoiceName: Stringa contenente il nome della voce di sistema da associare;
 - locale: Oggetto che rappresenta la lingua da utilizzare per la voce creata.
- **Descrizione:** Costruttore per gli oggetti di tipo MivoqVoice.

3.7.2.4.2 getEffects()

- **Accesso:** public;
- **Tipo di ritorno:** ArrayList<Effect>;
- **Descrizione:** Metodo che ritorna un array di Effect che verranno applicati alla voce.

3.7.2.4.3 getEncodedName(String gen, String myLang)

- **Accesso:** null;
- **Tipo di ritorno:** String;
- **Argomenti:**
 - gen: Genere della voce;
 - myLang: lingua della voce.
- **Descrizione:** Metodo che presi un genere e una voce, ritorna la voce di sistema da assegnare alla voce personalizzata.

3.7.2.4.4 getGender()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Metodo che ritorna il genere della voce.

3.7.2.4.5 getLanguage()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Metodo che ritorna la lingua della voce.

3.7.2.4.6 getName()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Metodo che ritorna il nome associato dall'utente alla voce.

3.7.2.4.7 getSampleText(String myLang)

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Argomenti:**
 - myLang: Stringa contenente la lingua che si vuole testare.
- **Descrizione:** Metodo statico che ritorna una riga di testo specifica per ogni lingua, utilizzabile per far testare all'utente la voce.

3.7.2.4.8 getStringEffects()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Metodo che ritorna la codifica in stringa di tutti gli effetti che dovranno essere applicati alla voce.

3.7.2.4.9 getVoiceName()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Metodo che ritorna il nome della voce di sistema che la voce personalizzata sta usando.

3.7.2.4.10 removeEffect(int index)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - index: intero che indica la posizione nell'array dell'effetto che deve essere rimosso.
- **Descrizione:** Metodo che permette di rimuovere un qualunque effetto da un voce.

3.7.2.4.11 setEffect(Effect e)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - e: Effetto da assegnare all'attuale voce.
- **Descrizione:** Metodo che permette di assegnare un nuovo effetto alla voce personalizzata.

3.7.2.4.12 setEmotion(Emotion e)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - e: Emozione da associare alla voce.
- **Descrizione:** Metodo che permette di associare un'emozione alla voce personalizzata.

3.7.2.4.13 setFemaleDe(boolean b)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - b: Variabile booleana che abilita o meno una correzione.
- **Descrizione:** Metodo che permette di utilizzare una voce tedesca di tipo femminile anche se non disponibile nelle voci di sistema.

3.7.2.4.14 setGender(String g)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - g: Stringa contenente il genere da assegnare ad una voce.
- **Descrizione:** Metodo che permette di modificare il genere di una voce.

3.7.2.4.15 setGenderLanguage(String gen, String myLang)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - gen: Stringa contenente il genere da assegnare alla voce;
 - myLang: Stringa contenente la lingua da assegnare alla voce;
- **Descrizione:** Metodo che permette di assegnare un genere ed una lingua ad una voce.

3.7.2.4.16 setName(String n)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - n: Stringa contenente il nuovo nome da assegnare.
- **Descrizione:** Metodo per la modifica del nome assegnato alla voce.

3.7.3 Language (class)

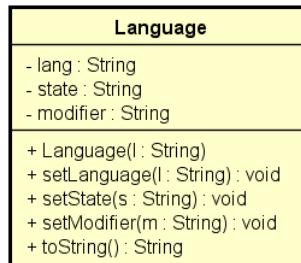


Figura 23: Classe Sivodim::Libraries::Model::Voice::Language

3.7.3.1 Descrizione

Classe che rappresenta i codici di lingua, paese e variante per la definizione del tipo di voce da utilizzare.

3.7.3.2 Utilizzo

Viene usata per rappresentare i codici di lingua, paese e variante per la definizione del tipo di voce.

3.7.3.3 Attributi

3.7.3.3.1 lang

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Attributo di identificazione di una specifica lingua.

3.7.3.3.2 state

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Attributo di memorizzazione di uno specifico stato per una lingua.

3.7.3.3.3 modifier

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Attributo di memorizzazione di una specifica variante per una lingua.

3.7.3.4 Metodi

3.7.3.4.1 Language(String l)

- **Accesso:** public;
- **Argomenti:**
 - l: Stringa contenente la lingua generica da applicare.
- **Descrizione:** Costruttore di oggetti di tipo Language.

3.7.3.4.2 setLanguage(String l)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - l: Stringa contenente la codifica della lingua generica da applicare.
- **Descrizione:** Metodo generico per l'impostazione della lingua.

3.7.3.4.3 setState(String s)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - s: Stringa contenente la codifica del modificatore statale per la lingua.
- **Descrizione:** Metodo che permette l'impostazione di un modificatore statale per una lingua.

3.7.3.4.4 setModifier(String m)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - m: Stringa contenente una variante per una lingua.
- **Descrizione:** Metodo che permette l'impostazione di una variante per la lingua.

3.7.3.4.5 toString()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Metodo che ritorna la codifica della lingua.

3.7.4 Emotion (class)

Emotion
- effects : String
+ Happiness : Emotion
+ Disgust : Emotion
+ Fear : Emotion
+ Sadness : Emotion
+ Surprise : Emotion
+ Anger : Emotion
+ Emotion(s : String)
+ toString() : String

Figura 24: Classe Sivodim::Libraries::Model::Voice::Emotion

3.7.4.1 Descrizione

Classe che rappresenta un insieme di effetti che definiscono un'emozione. Rende disponibile le emozioni rappresentandole come stringhe di testo statiche. Ogni stringa è pubblica e contiene un insieme di effetti (in forma testuale) applicabili a una voce di FA-TTS_G.

3.7.4.2 Utilizzo

Viene utilizzata per rappresentare le emozioni essere date ad una voce.

3.7.4.3 Attributi

3.7.4.3.1 effects

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Attributo stringa che contiene gli effetti da applicare alla voce.

3.7.4.3.2 Anger

- **Accesso:** public;
- **Tipo:** Emotion;
- **Descrizione:** Attributo statico per la rappresentazione dell'emozione rabbia.

3.7.4.3.3 Disgust

- **Accesso:** public;
- **Tipo:** Emotion;
- **Descrizione:** Attributo statico per la rappresentazione dell'emozione disgusto.

3.7.4.3.4 Fear

- **Accesso:** public;
- **Tipo:** Emotion;
- **Descrizione:** Attributo statico per la rappresentazione dell'emozione paura.

3.7.4.3.5 Happiness

- **Accesso:** public;
- **Tipo:** Emotion;
- **Descrizione:** Attributo statico per la rappresentazione dell'emozione felicità.

3.7.4.3.6 Sadness

- **Accesso:** public;
- **Tipo:** Emotion;
- **Descrizione:** Attributo statico per la rappresentazione dell'emozione tristezza.

3.7.4.3.7 Surprise

- **Accesso:** public;
- **Tipo:** Emotion;
- **Descrizione:** Attributo statico per la rappresentazione dell'emozione sorpresa.

3.7.4.4 Metodi

3.7.4.4.1 Emotion(String s)

- **Accesso:** public;
- **Argomenti:**
 - s: Stringa contenente la codifica dell'insieme di effetti che compongono un'emozione.
- **Descrizione:** Costruttore degli oggetti di tipo Emotion.

3.7.4.4.2 toString()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Metodo che ritorna la codifica dell'insieme di effetti che compongono un'emozione.

3.7.5 Effect (interface)

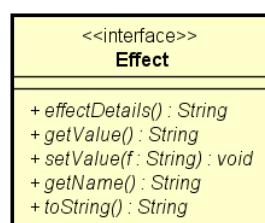


Figura 25: Classe Sivodim::Libraries::Model::Voice::Effect

3.7.5.1 Descrizione

Interfaccia che rappresenta il contratto attraverso cui implementare gli effetti di modifica della voce.

3.7.5.2 Utilizzo

Viene usata per definire il contratto della classe che rappresenta gli effetti di una voce.

3.7.5.3 Classi figlie

- Sivodim::Libraries::Model::Voice::EffectImpl.

3.7.5.4 Metodi

3.7.5.4.1 getName()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Metodo che ritorna il nome del effetto da applicare.

3.7.5.4.2 getValue()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Metodo che ritorna il valore del effetto che si vuole applicare.

3.7.5.4.3 setValue(String f)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - f: String contenente il valore da assegnare al effetto
- **Descrizione:** Metodo che permette di modificare il valore assegnato al effetto.

3.7.5.4.4 toString()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Metodo che ritorna la codifica dell'effetto da assegnare ad una voce.

3.7.6 EffectImpl (class)

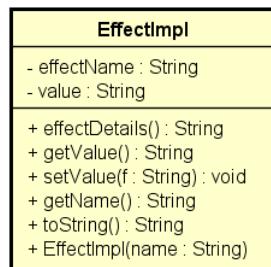


Figura 26: Classe Sivodim::Libraries::Model::Voice::EffectImpl

3.7.6.1 Descrizione

Classe che implementa l'interfaccia Effect.

3.7.6.2 Utilizzo

Implementa il contratto della classe Effect che gestisce gli effetti da dare ad una voce.

3.7.6.3 Classi ereditate

- Sivodim::Libraries::Model::Voice::Effect.

3.7.6.4 Attributi

3.7.6.4.1 effectName

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Attributo che memorizza il nome dell'effetto da applicare.

3.7.6.4.2 value

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Attributo che memorizza il valore dell'effetto da applicare.

3.7.6.5 Metodi

3.7.6.5.1 getName()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Implementazione del metodo dell'interfaccia, che ritorna il nome dell'effetto da applicare.

3.7.6.5.2 getValue()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Implementazione del metodo dell'interfaccia, che ritorna il valore da assegnare all'effetto che si vuole applicare.

3.7.6.5.3 setValue(String f)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - f: Stringa contenente il nuovo valore da assegnare all'effetto
- **Descrizione:** Implementazione del metodo dell'interfaccia, che permette di modificare il valore dell'effetto da applicare.

3.7.6.5.4 toString()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Implementazione del metodo dell'interfaccia, che ritorna la codifica in una stringa dell'effetto da applicare ad una voce.

3.7.6.5.5 EffectImpl(String name)

- **Accesso:** public;
- **Argomenti:**
 - name: Stringa contenente il nome del tipo di effetto da applicare.
- **Descrizione:** Costruttore degli oggetti EffectImpl.

3.8 Sivodim::Libraries::Presenter

3.8.1 Informazioni sul package

Di seguito viene illustrata, tramite un diagramma delle classi, la visione di insieme del package Libraries::Presenter contenente tutte le classi relative al Presenter nel rispetto del pattern MVP. Le classi Presenter possono avere più di un riferimento alle classi View, per tanto il rapporto fra View e Presenter non è 1:1. Ogni Presenter è progettato con l'intento di garantire la persistenza dei dati in memoria in riferimento alle componenti del Model. In questo modo è possibile garantire che le operazioni sulle View non interferiscano con il Model. I Presenter non sono attributi di classe, ma continuano a esistere al di fuori delle View a cui sono associati: per tale ragione i loro riferimenti devono essere dichiarati come attributi statici.

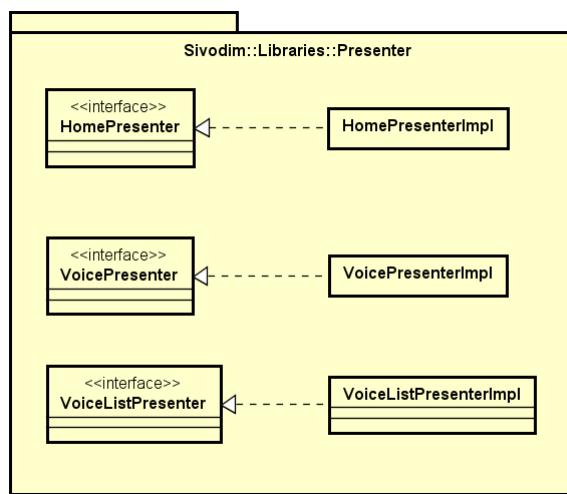


Figura 27: Componente Sivodim::Libraries::Presenter

3.8.2 HomePresenter (interface)



Figura 28: Classe Sivodim::Libraries::Presenter::HomePresenter

3.8.2.1 Descrizione

Interfaccia che definisce il contratto attraverso cui implementare il *presenter* della pagina iniziale dell'applicazione di configurazione del motore di sintesi.

3.8.2.2 Utilizzo

Viene usata per definire il contratto di implementazione del *presenter* associato alla HomeActivity contenuta nella componente Sivodim::Libraries::View.

3.8.2.3 Classi figlie

- Sivodim::Libraries::Presenter::HomePresenterImpl.

3.8.2.4 Metodi

3.8.2.4.1 goToVoiceListActivity(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Sposta la view portando attiva la VoiceListActivity dopo aver istanziato un VoiceListPresenter.

3.8.2.4.2 goToNewVoiceActivity(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Sposta la view portando attiva la NewVoiceActivity dopo aver istanziato un VoicePresenter.

3.8.3 HomePresenterImpl (class)

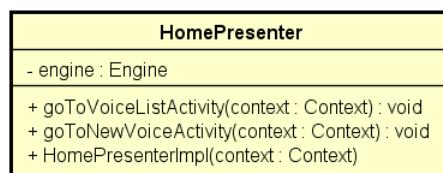


Figura 29: Classe Sivodim::Libraries::Presenter::HomePresenterImpl

3.8.3.1 Descrizione

Classe che implementa l'interfaccia HomePresenter.

3.8.3.2 Utilizzo

Viene utilizzata per implementare l'interfaccia HomePresenter.

3.8.3.3 Classi ereditate

- Sivodim::Libraries::Presenter::HomePresenter.

3.8.3.4 Attributi

3.8.3.4.1 engine

- **Accesso:** private;
- **Tipo:** Engine;
- **Descrizione:** Attributo che memorizza l'istanza del motore di sintesi vocale, permettendo la sintesi sia in presenza di connessione, mediante al servizio offerto dai server di MIVOQ s.r.l., sia in caso di assenza, mediante il servizio TTS_G di Android_G.

3.8.3.5 Metodi

3.8.3.5.1 HomePresenterImpl(Context context)

- **Accesso:** public;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Costruttore che prende per parametro un contesto e istanzia un Engine per la sintesi vocale. HomePresenterImpl è utilizzata per la gestione dell'interfaccia utente, nel rispetto del pattern MVP.

3.8.3.5.2 goToNewVoiceActivity(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Istanzia un VoicePresenter. Successivamente assegna al campo statico di NewVoiceActivity questo presenter e, attraverso una Intent, porta l'utente in questa Activity.

3.8.3.5.3 goToVoiceListActivity(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view.

- **Descrizione:** Istanzia un VoiceListPresenter. Successivamente assegna al campo statico di VoiceListActivity questo presenter e, attraverso una Intent, porta l'utente in questa Activity.

3.8.4 VoicePresenter (interface)

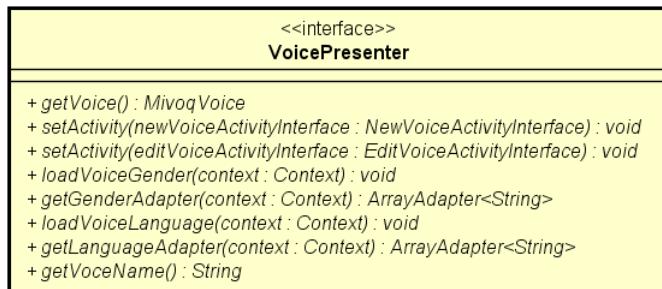


Figura 30: Classe Sivodim::Libraries::Presenter::VoicePresenter

3.8.4.1 Descrizione

Interfaccia che definisce il contratto attraverso cui implementare il *presenter* della pagina relativa alla gestione delle voci.

3.8.4.2 Utilizzo

Viene usata per definire il contratto di implementazione del *presenter* associato alle NewVoiceActivity, VoiceListActivity, EditVoiceActivity presenti nella componente Sivodim::Libraries::View.

3.8.4.3 Classi figlie

- Sivodim::Libraries::Presenter::VoicePresenterImpl.

3.8.4.4 Metodi

3.8.4.4.1 setActivity(NewVoiceActivityInterface newVoiceActivityInterface)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - newVoiceActivityInterface: interfaccia dell'activity che si occupa della creazione di una nuova voce.
- **Descrizione:** Registra una NewVoiceActivityInterface al presenter, in accordo al modello MVP.

3.8.4.4.2 setActivity(EditVoiceActivityInterface newVoiceActivityInterface)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - editVoiceActivityInterface: interfaccia dell'activity che si occupa della gestione della modifica delle voci.
- **Descrizione:** Registra una EditVoiceInterface al presenter, in accordo al modello MVP.

3.8.4.4.3 loadVoiceGender(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Metodo che si occupa della creazione di un ArrayAdapter contenente i possibili generi delle lingue offerte dal servizio di sintesi di MIVOQ s.r.l..

3.8.4.4.4 getGenderAdapter(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** ArrayAdapter<String>;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Metodo che crea un ArrayAdapter con i generi delle voci presenti in archivio.

3.8.4.4.5 loadVoiceLanguage(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Metodo che crea un ArrayAdapter con le lingue delle voci presenti in archivio.

3.8.4.4.6 getLanguageAdapter(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** ArrayAdapter<String>;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Metodo che ritorna un ArrayAdapter di tipo String con memorizzate le lingue delle voci disponibili.

3.8.4.4.7 getVoiceName()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Ritorna la Stringa con il nome della voce di tipo MivoqVoice.

3.8.5 VoicePresenterImpl (class)

VoicePresenterImpl	
- mivoqVoice : MivoqVoice	
- engine : Engine	
- genderAdapter : ArrayAdapter<String>	
- languageAdapter : ArrayAdapter<String>	
- editVoiceActivityInterface : EditVoiceActivityInterface	
- newVoiceActivityInterface : NewVoiceActivityInterface	
+ getVoice() : MivoqVoice	
+ setActivity(newVoiceActivityInterface : NewVoiceActivityInterface) : void	
+ setActivity(editVoiceActivityInterface : EditVoiceActivityInterface) : void	
+ loadVoiceGender(context : Context) : void	
+ getGenderAdapter(context : Context) : ArrayAdapter<String>	
+ loadVoiceLanguage(context : Context) : void	
+ getLanguageAdapter(context : Context) : ArrayAdapter<String>	
+ getVoiceName() : String	
+ VoicePresenterImpl(engine : Engine) : void	
+ VoicePresenterImpl(mivoqVoice : MivoqVoice) : void	

Figura 31: Classe Sivodim::Libraries::Presenter::VoicePresenterImpl

3.8.5.1 Descrizione

Classe che implementa l'interfaccia VoicePresenter.

3.8.5.2 Utilizzo

Viene usata per implementare il contratto del *presenter* associato alle NewVoiceActivity, VoiceListActivity, EditVoiceActivity presenti nella componente Sivodim::Libraries::View.

3.8.5.3 Classi ereditate

- Sivodim::Libraries::Presenter::VoicePresenter.

3.8.5.4 Attributi

3.8.5.4.1 editVoiceActivityInterface

- **Accesso:** private;
- **Tipo:** EditVoiceActivityInterface;
- **Descrizione:** Riferimento alla view EditVoiceActivityInterface che VoicePresenterImpl utilizza per interagire con l'interfaccia utente nella modifica della voce, nel rispetto del pattern MVP.

3.8.5.4.2 engine

- **Accesso:** private;
- **Tipo:** Engine;
- **Descrizione:** Attributo che memorizza l'istanza del motore di sintesi vocale, permettendo la sintesi sia in presenza di connessione, mediante il servizio offerto dai server di MIVOQ s.r.l., sia in caso di assenza, mediante il servizio TTS_G di Android_G.

3.8.5.4.3 genderAdapter

- **Accesso:** private;
- **Tipo:** ArrayAdapter<String>;
- **Descrizione:** Riferimento ad ArrayAdapter (prelevato dal VoicePresenter) da associare alla ListView per visualizzare dinamicamente nella GUI i dati relativi ai generi disponibili per ciascuna voce.

3.8.5.4.4 languageAdapter

- **Accesso:** private;
- **Tipo:** ArrayAdapter<String>;
- **Descrizione:** Riferimento ad ArrayAdapter (prelevato dal VoicePresenter) da associare alla ListView per visualizzare dinamicamente nella GUI i dati relativi alle lingue disponibili per ciascuna voce.

3.8.5.4.5 mivoqVoice

- **Accesso:** private;
- **Tipo:** MivoqVoice;
- **Descrizione:** Attributo usato per salvare un'istanza della voce MivoqVoice da passare all'EditVoiceActivity per la modifica della voce.

3.8.5.4.6 newVoiceActivityInterface

- **Accesso:** private;
- **Tipo:** NewVoiceActivityInterface;
- **Descrizione:** Riferimento alla view NewVoiceActivityInterface che VoicePresenterImpl utilizza per interagire con l'interfaccia utente nella creazione di una voce, nel rispetto del pattern MVP.

3.8.5.5 Metodi

3.8.5.5.1 VoicePresenterImpl(Engine engine)

- **Accesso:** public;
- **Argomenti:**
 - engine: istanza del motore di sintesi vocale.
- **Descrizione:** Costruttore che prendere come parametro un riferimento all'Engine e costruisce un oggetto di tipo VoicePresenterImpl.

3.8.5.5.2 VoicePresenterImpl(Mivoq mivoq)

- **Accesso:** public;
- **Argomenti:**
 - mivoqVoice: oggetto di tipo MivoqVoice, che rappresenta una voce a cui si possono applicare gli effetti proposti dal servizio Mivoq.
- **Descrizione:** Secondo costruttore per la classe VoicePresenterImpl, che dato un oggetto di tipo MivoqVoice crea un VoicePresenterImpl.

3.8.5.5.3 setActivity(NewVoiceActivityInterface newVoiceActivityInterface)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - newVoiceActivityInterface: interfaccia dell'activity che si occupa della creazione di una nuova voce.
- **Descrizione:** Memorizza un riferimento a NewVoiceActivityInterface, che utilizza per interagire con l'interfaccia utente, nel rispetto del pattern MVP.

3.8.5.5.4 setActivity(EditVoiceActivityInterface newVoiceActivityInterface)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - editVoiceActivityInterface: interfaccia dell'activity che si occupa della gestione della modifica delle voci.
- **Descrizione:** Secondo metodo setActivity che si occupa di memorizzare un riferimento a EditVoiceActivityInterface, che utilizza per interagire con l'interfaccia utente, nel rispetto del pattern MVP.

3.8.5.5.5 loadVoiceGender(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Implementazione della classe che si occupa della creazione di un ArrayAdapter contenente i possibili generi delle lingue offerte dal servizio di sintesi di MIVOQ s.r.l..

3.8.5.5.6 getGenderAdapter(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** ArrayAdapter<String>;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Implementazione del metodo che crea un ArrayAdapter con i generi delle voci presenti in archivio.

3.8.5.5.7 loadVoiceLanguage(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Implementazione del metodo che crea un ArrayAdapter con le lingue delle voci presenti in archivio.

3.8.5.5.8 getLanguageAdapter(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** ArrayAdapter<String>;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Implementazione del metodo che ritorna un ArrayAdapter di tipo String con memorizzate le lingue delle voci disponibili.

3.8.5.5.9 getVoiceName()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Ritorna la Stringa con il nome della voce di tipo MivoqVoice.

3.8.6 VoiceListPresenter (interface)



Figura 32: Classe Sivodim::Libraries::Presenter::VoiceListPresenter

3.8.6.1 Descrizione

Interfaccia che rappresenta il contratto attraverso cui implementare il *presenter* della pagina relativa alla gestione della lista delle voci.

3.8.6.2 Utilizzo

Viene usata per definire il contratto di implementazione del *presenter* associato alla `VoiceListActivity` presenti nella componente `Sivodim::Libraries::View`.

3.8.6.3 Classi figlie

- `Sivodim::Libraries::Presenter::VoiceListPresenterImpl`.

3.8.6.4 Metodi

3.8.6.4.1 goToEditVoiceActivity(Context context, MivoqVoice mivoqVoice)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view;
 - mivoqVoice: la voce che viene passata alla EditVoiceActivity, per la sua modifica.
- **Descrizione:** Sposta la view portando attiva la EditVoiceActivity dopo aver istanziato un VoicePresenter, costruito con il riferimento alla voce passato come parametro.

3.8.6.4.2 setActivity(VoiceListActivityInterface voiceListActivityInterface)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - voiceListActivityInterface: interfaccia dell'activity che si occupa della gestione della lista delle voci.
- **Descrizione:** Registra una EditSpeechInterface al presenter, in accordo al modello MVP.

3.8.6.4.3 createMivoqVoice(String s)

- **Accesso:** public;
- **Tipo di ritorno:** MivoqVoice;
- **Argomenti:**
 - s: stringa che indica il nome della voce.
- **Descrizione:** Metodo che si occupa di creare una MivoqVoice dall'Engine di sintesi vocale.

3.8.6.4.4 loadVoiceNames(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Metodo che crea un ArrayAdapter con le voci presenti in archivio.

3.8.6.4.5 getVoicesAdapter(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** ArrayAdapter<String>;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Ritorna un ArrayAdapter di tipo String con memorizzate i nomi delle voci.

3.8.6.4.6 goToEditVoiceActivity(Context context, MivoqVoice mivoqVoice)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view;
 - mivoqVoice: la voce che viene passata alla EditVoiceActivity, per la sua modifica.
- **Descrizione:** Istanzia un VoicePresenter. Successivamente assegna al campo statico di EditVoiceActivity questo presenter e, attraverso una Intent, porta l'utente in questa Activity.

3.8.7 VoiceListPresenterImpl (class)

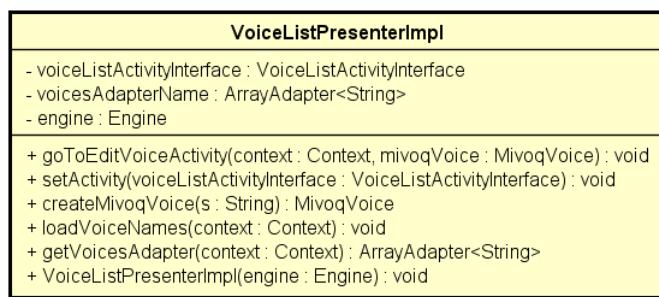


Figura 33: Classe Sivodim::Libraries::Presenter::VoiceListPresenterImpl

3.8.7.1 Descrizione

Classe che implementa l'interfaccia VoiceListPresenter.

3.8.7.2 Utilizzo

Viene usata per implementare il contratto del *presenter* associato alle VoiceListActivity presenti nella componente Sivodim::Libraries::View.

3.8.7.3 Classi ereditate

- Sivodim::Libraries::Presenter::VoiceListPresenter.

3.8.7.4 Attributi

3.8.7.4.1 engine

- **Accesso:** private;
- **Tipo:** Engine;
- **Descrizione:** Attributo che memorizza l'istanza del motore di sintesi vocale, permettendo la sintesi sia in presenza di connessione, mediante al servizio offerto dai server di MIVOQ s.r.l., sia in caso di assenza, mediante il servizio TTS_G di Android_G.

3.8.7.4.2 voiceListActivityInterface

- **Accesso:** private;
- **Tipo:** VoiceListActivityInterface;
- **Descrizione:** Riferimento alla view VoiceListActivityInterface che VoiceListPresenterImpl utilizza per interagire con l'interfaccia utente, nel rispetto del pattern MVP.

3.8.7.4.3 voicesAdapterName

- **Accesso:** private;
- **Tipo:** ArrayAdapter<String>;
- **Descrizione:** Riferimento ad ArrayAdapter (prelevato dal VoiceListPresenter) da associare alla ListView per visualizzare dinamicamente nella GUI i dati relativi alle voci presenti in archivio.

3.8.7.5 Metodi

3.8.7.5.1 VoiceListPresenterImpl(Engine engine)

- **Accesso:** public;
- **Argomenti:**
 - engine: il motore di sintesi vocale.
- **Descrizione:** Costruttore che dato un oggetto di tipo Engine si occupa di creare un oggetto della classe VoiceListPresenterImpl.

3.8.7.5.2 setActivity(VoiceListActivityInterface voiceListActivityInterface)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - voiceListActivityInterface: interfaccia dell'activity che si occupa della gestione della lista delle voci.
- **Descrizione:** Memorizza un riferimento a VoiceListInterface, che utilizza per interagire con l'interfaccia utente, nel rispetto del pattern MVP.

3.8.7.5.3 createMivoqVoice(String s)

- **Accesso:** public;
- **Tipo di ritorno:** MivoqVoice;
- **Argomenti:**
 - s: stringa che indica il nome della voce.
- **Descrizione:** Data una stringa ritorna una voce dell'engine.

3.8.7.5.4 loadVoiceNames(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Implementazione del metodo che crea un ArrayAdapter con le voci presenti in archivio.

3.8.7.5.5 getVoicesAdapter(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** ArrayAdapter<String>;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Implementazione del metodo che ritorna un ArrayAdapter di tipo String con memorizzati i nomi delle voci.

3.9 Sivodim::Libraries::View

3.9.1 Informazioni sul package

Di seguito viene illustrata, tramite un diagramma delle classi, la visione di insieme del package Libraries::View contenente tutte le classi relative al View nel rispetto del pattern MVP.

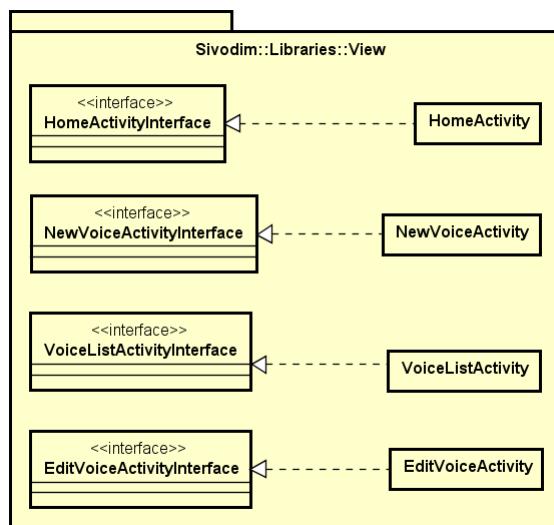


Figura 34: Componente Sivodim::Libraries::View

3.9.2 HomeActivityInterface (interface)



Figura 35: Classe Sivodim::Libraries::View::HomeActivityInterface

3.9.2.1 Descrizione

Interfaccia che rappresenta il contratto attraverso cui implementare la view della pagina principale dell'applicazione e catturarne gli eventi.

3.9.2.2 Utilizzo

Viene usata per definire il contratto di interazione con l'utente nella pagina principale dell'applicazione.

3.9.2.3 Classi figlie

- Sivodim::Libraries::View::HomeActivity.

3.9.3 HomeActivity (class)

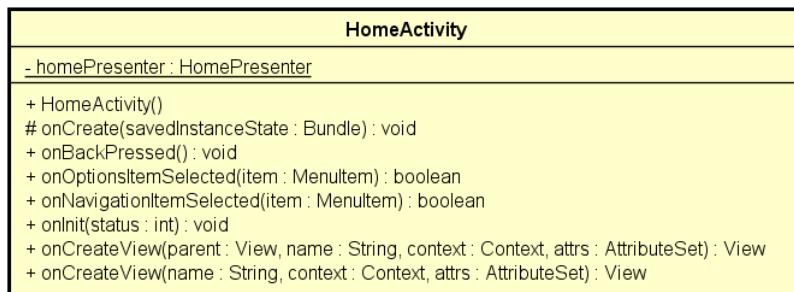


Figura 36: Classe Sivodim::Libraries::View::HomeActivity

3.9.3.1 Descrizione

Classe che implementa l'interfaccia HomeActivity.

3.9.3.2 Utilizzo

Implementa il contratto di interazione con l'utente nella pagina principale dell'applicazione.

3.9.3.3 Classi ereditate

- Sivodim::Libraries::View::HomeActivityInterface.

3.9.3.4 Attributi

3.9.3.4.1 homePresenter

- **Accesso:** private;
- **Tipo:** HomePresenter;
- **Descrizione:** Riferimento statico al HomePresenter che la view HomeActivity utilizza per interagire con il model, nel rispetto del pattern MVP.

3.9.4 NewVoiceActivityInterface (interface)



Figura 37: Classe Sivodim::Libraries::View::NewVoiceActivityInterface

3.9.4.1 Descrizione

Interfaccia che rappresenta il contratto attraverso cui implementare la view attraverso cui l'utente può creare una nuova voce personalizzata.

3.9.4.2 Utilizzo

Viene usata per definire il contratto di interazione con l'utente nella pagina dedicata alla creazione di una nuova voce personalizzata.

3.9.4.3 Classi figlie

- Sivodim::Libraries::View::NewVoiceActivity.

3.9.5 NewVoiceActivity (class)

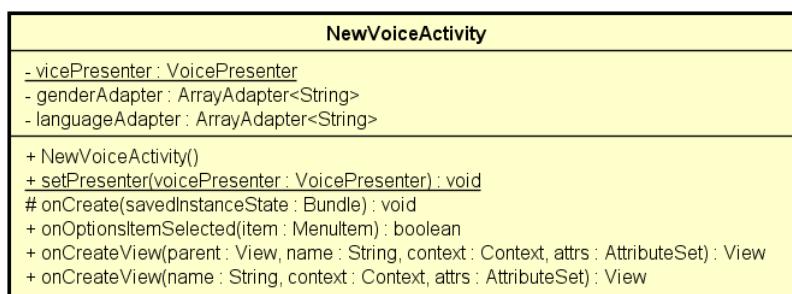


Figura 38: Classe Sivodim::Libraries::View::NewVoiceActivity

3.9.5.1 Descrizione

Classe che implementa l'interfaccia NewVoiceActivity.

3.9.5.2 Utilizzo

Implementa il contratto di interazione con l'utente nella pagina dedicata alla creazione di nuove voci personalizzate.

3.9.5.3 Classi ereditate

- Sivodim::Libraries::View::NewVoiceActivityInterface.

3.9.5.4 Attributi

3.9.5.4.1 genderAdapter

- **Accesso:** private;
- **Tipo:** ArrayAdapter<String>;
- **Descrizione:** Riferimento ad ArrayAdapter (prelevato dal VoicePresenter e visualizzato nella NewVoiceActivity) da associare alla ListView per visualizzare dinamicamente nella GUI i dati relativi ai generi disponibili per ciascuna voce.

3.9.5.4.2 languageAdapter

- **Accesso:** private;
- **Tipo:** ArrayAdapter<String>;
- **Descrizione:** Riferimento ad ArrayAdapter (prelevato dal VoicePresenter e visualizzato nella NewVoiceActivity) da associare alla ListView per visualizzare dinamicamente nella GUI i dati relativi alle lingue disponibili per ciascuna voce.

3.9.5.4.3 voicePresenter

- **Accesso:** private;
- **Tipo:** VoicePresenter;
- **Descrizione:** Riferimento statico al VoicePresenter che la view NewVoiceActivity utilizza per interagire con il model, nel rispetto del pattern MVP.

3.9.6 EditVoiceActivityInterface (interface)



Figura 39: Classe Sivodim::Libraries::View::EditVoiceActivityInterface

3.9.6.1 Descrizione

Interfaccia che rappresenta il contratto attraverso cui implementare la view della pagina dedicata alla modifica di una voce già esistente.

3.9.6.2 Utilizzo

Viene usata per definire il contratto di interazione con l'utente nella pagina dedicata alla modifica di una voce già esistente.

3.9.6.3 Classi figlie

- Sivodim::Libraries::View::EditVoiceActivity.

3.9.7 EditVoiceActivity (class)

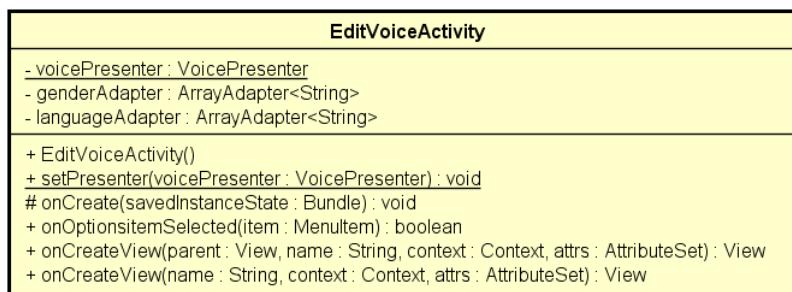


Figura 40: Classe Sivodim::Libraries::View::EditVoiceActivity

3.9.7.1 Descrizione

Classe che implementa l'interfaccia EditVoiceActivity.

3.9.7.2 Utilizzo

Implementa il contratto di interazione con l'utente nella pagina dedicata alla modifica di una voce già esistente.

3.9.7.3 Classi ereditate

- Sivodim::Libraries::View::EditVoiceActivityInterface.

3.9.7.4 Attributi

3.9.7.4.1 genderAdapter

- **Accesso:** private;
- **Tipo:** ArrayAdapter<String>;
- **Descrizione:** Riferimento ad ArrayAdapter (prelevato dal VoicePresenter e visualizzato nella EditVoiceActivity) da associare alla ListView per visualizzare dinamicamente nella GUI i dati relativi ai generi disponibili per ciascuna voce.

3.9.7.4.2 languageAdapter

- **Accesso:** private;
- **Tipo:** ArrayAdapter<String>;
- **Descrizione:** Riferimento ad ArrayAdapter (prelevato dal VoicePresenter e visualizzato nella EditVoiceActivity) da associare alla ListView per visualizzare dinamicamente nella GUI i dati relativi alle lingue disponibili per ciascuna voce.

3.9.7.4.3 voicePresenter

- **Accesso:** private;
- **Tipo:** VoiceListActivityInterface;
- **Descrizione:** Riferimento statico al VoicePresenter che la view EditVoiceActivity utilizza per interagire con il model, nel rispetto del pattern MVP.

3.9.8 VoiceListActivityInterface (interface)



Figura 41: Classe Sivodim::Libraries::View::VoiceListActivityInterface

3.9.8.1 Descrizione

Interfaccia che rappresenta il contratto attraverso cui implementare la view della pagina contenente la lista completa delle voci offerte dall'applicazione.

3.9.8.2 Utilizzo

Viene usata per definire il contratto di interazione con l'utente nella pagina contenente la lista completa delle voci offerte dall'applicazione.

3.9.8.3 Classi figlie

- Sivodim::Libraries::View::VoiceListActivity.

3.9.9 VoiceListActivity (class)

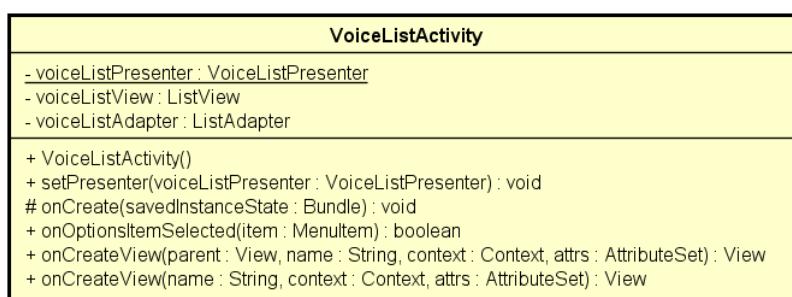


Figura 42: Classe Sivodim::Libraries::View::VoiceListActivity

3.9.9.1 Descrizione

Classe che implementa l'interfaccia VoiceListActivity.

3.9.9.2 Utilizzo

Implementa il contratto di interazione con l'utente nella pagina contenente la lista completa delle voci offerte dall'applicazione.

3.9.9.3 Classi ereditate

- Sivodim::Libraries::View::VoiceListActivityInterface.

3.9.9.4 Attributi

3.9.9.4.1 voiceListAdapter

- **Accesso:** private;
- **Tipo:** ListAdapter;
- **Descrizione:** Riferimento a ListAdapter (prelevato dal VoiceListPresenter e visualizzato nella VoiceListActivity) da associare alla ListView per visualizzare dinamicamente nella GUI i dati relativi alle voci disponibili.

3.9.9.4.2 voiceListPresenter

- **Accesso:** private;
- **Tipo:** VoiceListPresenter;
- **Descrizione:** Riferimento statico al VoiceListPresenter che la view VoiceListActivity utilizza per interagire con il model, nel rispetto del pattern MVP .

3.9.9.4.3 voiceListView

- **Accesso:** private;
- **Tipo:** ListView;
- **Descrizione:** Componente grafica che permette di visualizzare la lista di tutte le lingua salvate in archivio.

3.10 Sivodim::Drama

3.10.1 Informazioni sul package

Di seguito viene illustrata, tramite un diagramma delle classi, la visione di insieme del package Drama contenente tutte le classi relative all'applicazione Speeches (app dedicata alla creazione di sceneggiati).

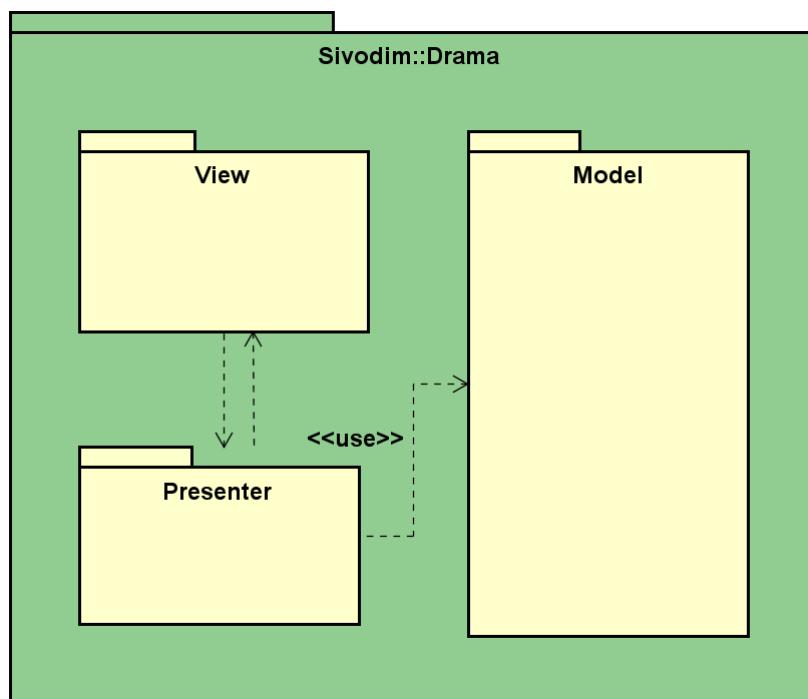


Figura 43: Componente Sivodim::Drama

3.11 Sivodim::Drama::Model

3.11.1 Informazioni sul package

Di seguito viene illustrata, tramite un diagramma delle classi, la visione di insieme del package Drama::Model contenente tutte le classi relative al Model nel rispetto del pattern MVP.

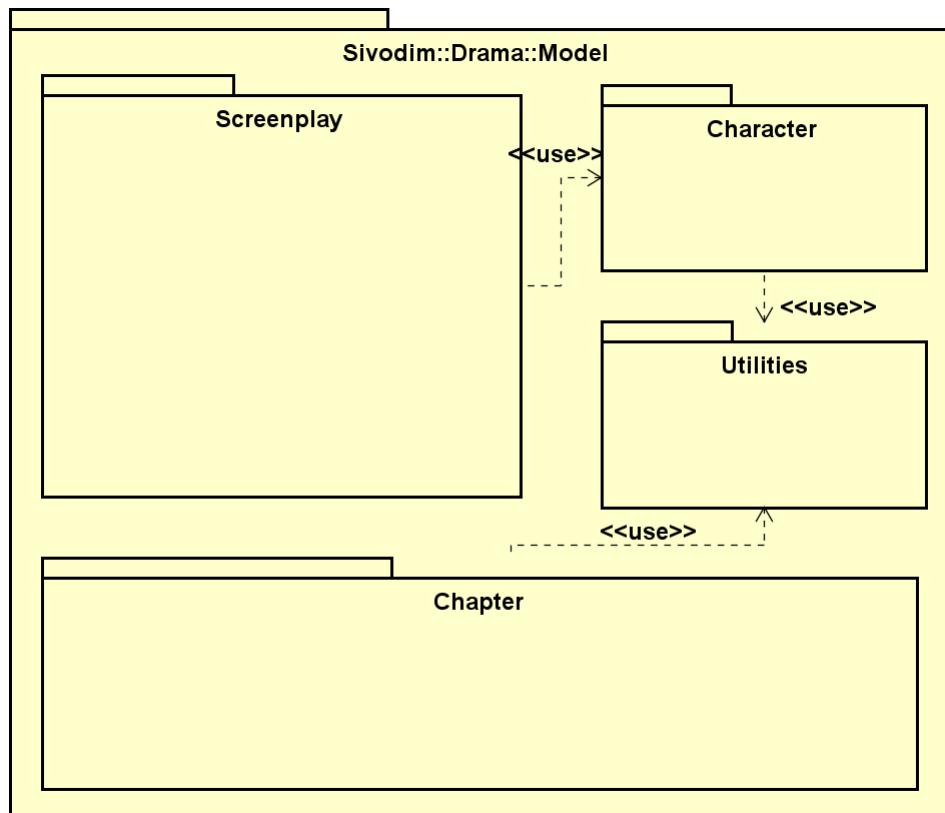


Figura 44: Componente Sivodim::Drama::Model

3.12 Sivodim::Drama::Presenter

3.12.1 Informazioni sul package

Di seguito viene illustrata, tramite un diagramma delle classi, la visione di insieme del package Drama::Presenter contenente tutte le classi relative al Presenter nel rispetto del pattern MVP. Le classi Presenter possono avere più di un riferimento alle classi View, per tanto il rapporto fra View e Presenter non è 1:1. Ogni Presenter è progettato con l'intento di garantire la persistenza dei dati in memoria in riferimento alle componenti del Model. In questo modo è possibile garantire che le operazioni sulle View non interferiscano con il Model. I Presenter non sono attributi di classe, ma continuano a esistere al di fuori delle View a cui sono associati: per tale ragione i loro riferimenti devono essere dichiarati come attributi statici.

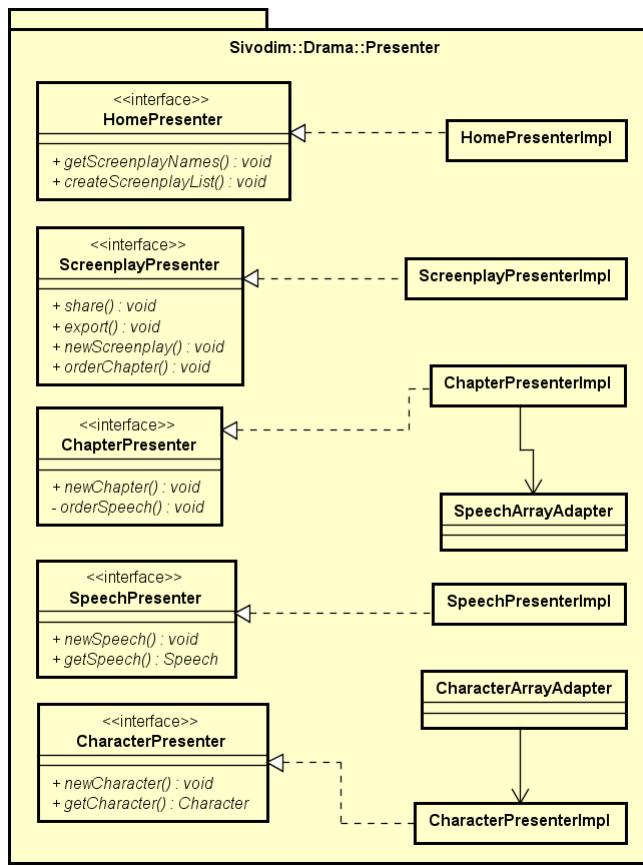


Figura 45: Componente Sivodim::Drama::Presenter

3.12.2 HomePresenter (interface)

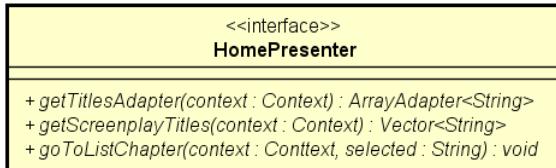


Figura 46: Classe Sivodim::Drama::Presenter::HomePresenter

3.12.2.1 Descrizione

Interfaccia che rappresenta il contratto attraverso cui implementare il *presenter* della pagina iniziale dell'applicazione.

3.12.2.2 Utilizzo

Viene usata per definire il contratto di implementazione del *presenter* associato alla HomeActivity presente nella componente Sivodim::Drama::View. Tale presenter dovrà gestire la vista principale dell'applicazione e fornirne metodi per la navigazione e il recupero di informazioni.

3.12.2.3 Classi figlie

- Sivodim::Drama::Presenter::HomePresenterImpl.

3.12.2.4 Metodi

3.12.2.4.1 getTitlesAdapter(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** ArrayAdapter<String>;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Restituisce un ArrayAdapter di String con i nomi degli Screenplay salvati in memoria;
- **Note:** Questo metodo non istanzia alcuno Screenplay, si limita a trovarne i nomi.

3.12.2.4.2 getScreenplayTitles(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** Vector<String>;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Restituisce un Vector di String con i nomi degli Screenplay salvati in memoria senza caricarli.

3.12.2.4.3 goToListChapter(Context context, String selected)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view;
 - selected: il nome che identifica lo Screenplay da aprire.
- **Descrizione:** Sposta la view portando attiva la ListChapterActivity dopo aver istanziato uno ScreenplayPresenter sullo Screenplay indicato tramite la Stringa con il nome passata per parametro.

3.12.3 HomePresenterImpl (class)

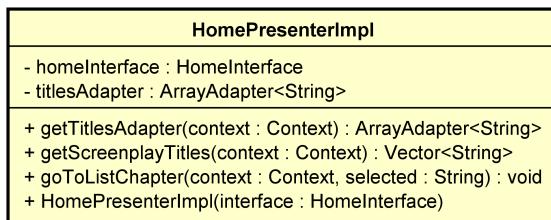


Figura 47: Classe Sivodim::Drama::Presenter::HomePresenterImpl

3.12.3.1 Descrizione

Classe che implementa l'interfaccia HomePresenter.

3.12.3.2 Utilizzo

Viene usata per implementare l'interfaccia HomePresenter.

3.12.3.3 Classi ereditate

- Sivodim::Drama::Presenter::HomePresenter.

3.12.3.4 Attributi

3.12.3.4.1 titleAdapter

- **Accesso:** private;
- **Tipo:** ArrayAdapter<String>;
- **Descrizione:** ArrayAdapter contenente una lista dinamica di stringhe relative ai titoli di ogni sceneggiato: utile per non caricare in memoria tutti gli sceneggiati all'avvio dell'applicazione. Ogni titolo va associato (nella HomeActivity) a un oggetto Button che se premuto porta all'interno di uno specifico sceneggiato.

3.12.3.4.2 homeInterface

- **Accesso:** private;
- **Tipo:** HomeInterface;
- **Descrizione:** Riferimento alla view HomeInterface che HomePresenterImpl utilizza per interagire con l'interfaccia utente, nel rispetto del pattern MVP.

3.12.3.5 Metodi

3.12.3.5.1 HomePresenterImpl(HomeInterface homeInterface)

- **Accesso:** public;
- **Argomenti:**
 - homeInterface: l'activity che utilizza questo presenter.
- **Descrizione:** Costruttore che prende per parametro un riferimento alla HomeInterface, che HomePresenterImpl utilizza per interagire con l'interfaccia utente, nel rispetto del pattern MVP.

3.12.3.5.2 getScreenplayTitles(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** Vector<String>;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia HomePresenterImpl. Carica dalla memoria i titoli degli sceneggiati presenti, senza però fare il load degli sceneggiati e li restituisce in un Vector.

3.12.3.5.3 getTitlesAdapter(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** ArrayAdapter<String>;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia HomePresenterImpl. Restituisce un ArrayAdapter di String con i titoli degli sceneggiati presenti in memoria, utilizzando il metodo getScreenplayTitles.

3.12.3.5.4 goToListChapter(Context context, String selected)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view;
 - selected: il titolo dello Screenplay da aprire.

- **Descrizione:** Istanzia uno ScreenplayPresenter passando per parametro uno Screenplay caricato dalla memoria (utilizzando il metodo statico Screenplay.loadScreenplay) con il nome indicato dal parametro selected. Successivamente assegna al campo statico di ListChapterActivity questo presenter e, attraverso una Intent porta l'utente in questa Activity.

3.12.4 ScreenplayPresenter (interface)

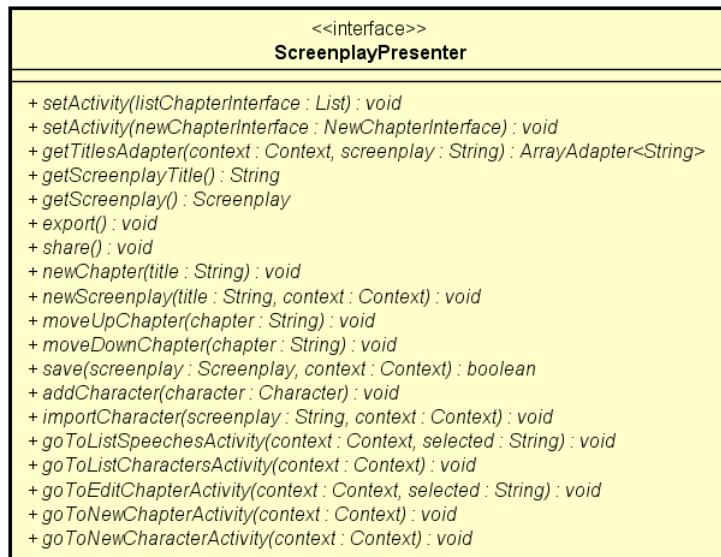


Figura 48: Classe Sivodim::Drama::Presenter::ScreenplayPresenter

3.12.4.1 Descrizione

Interfaccia che rappresenta il contratto attraverso cui implementare il *presenter* delle pagine relative alla gestione degli sceneggiati.

3.12.4.2 Utilizzo

Viene usata per definire il contratto di implementazione del *presenter* associato alle ListChapterActivity, ShareActivity, ExportActivity e NewScreenplayActivity presenti nella componente Sivodim::Drama::View.

3.12.4.3 Classi figlie

- Sivodim::Drama::Presenter::ScreenplayPresenterImpl.

3.12.4.4 Metodi

3.12.4.4.1 setActivity(ListChapterInterface listChapterInterface)

- **Accesso:** public;
- **Tipo di ritorno:** void;

- **Argomenti:**

- listChapterInterface: riferimento alla activity che utilizza il presenter.

- **Descrizione:** Memorizza un riferimento a ListChapterInterface, che utilizza per interagire con l'interfaccia utente, nel rispetto del pattern MVP.

3.12.4.4.2 setActivity(NewChapterInterface newChapterInterface)

- **Accesso:** public;

- **Tipo di ritorno:** void;

- **Argomenti:**

- newChapterInterface: riferimento alla activity che utilizza il presenter.

- **Descrizione:** Memorizza un riferimento a NewChapterInterface, che utilizza per interagire con l'interfaccia utente, nel rispetto del pattern MVP.

3.12.4.4.3 getTitlesAdapter(Context context, String screenplay)

- **Accesso:** public;

- **Tipo di ritorno:** ArrayAdapter<String>;

- **Argomenti:**

- context: il contesto della view;

- screenplay: lo Screenplay di cui si vogliono conoscere i capitoli.

- **Descrizione:** Restituisce un ArrayAdapter di String con i nomi dei capitoli dello Screenplay.

3.12.4.4.4 getScreenplayTitle()

- **Accesso:** public;

- **Tipo di ritorno:** String;

- **Descrizione:** Restituisce il titolo dello Screenplay.

3.12.4.4.5 getScreenplay()

- **Accesso:** public;

- **Tipo di ritorno:** Screenplay;

- **Descrizione:** Restituisce lo screenplay memorizzato nel presenter.

3.12.4.4.6 `export()`

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Descrizione:** Richiama le funzionalità di esportazione presenti nello Screenplay memorizzato nel proprio stato.

3.12.4.4.7 `share()`

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Descrizione:** Richiama le funzionalità di condivisione presenti nello Screenplay memorizzato nel proprio stato.

3.12.4.4.8 `newChapter(String title)`

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Descrizione:** Crea un capitolo con il titolo ricevuto per parametro e lo aggiunge allo Screenplay.

3.12.4.4.9 `newScreenplay(String title, Context context)`

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - title: il titolo dello sceneggiato da creare;
 - context: il contesto della view.
- **Descrizione:** Crea un nuovo Screenplay e lo salva in memoria.

3.12.4.4.10 `moveUpChapter(String chapterTitle)`

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - chapterTitle: il titolo del capitolo da spostare.
- **Descrizione:** Sposta indietro nell'ordine dei capitoli un capitolo dello Screenplay, identificato dal nome passato come String.

3.12.4.4.11 moveDownChapter(String chapterTitle)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - chapterTitle: il titolo del capitolo da spostare.
- **Descrizione:** Sposta in avanti nell'ordine dei capitoli un capitolo dello Screenplay, identificato dal nome passato come String.

3.12.4.4.12 save(Screenplay screenplay, Context context)

- **Accesso:** public;
- **Tipo di ritorno:** boolean;
- **Argomenti:**
 - screenplay: lo sceneggiato da salvare;
 - context: il contesto dello sceneggiato.
- **Descrizione:** Salva lo Screenplay in memoria sotto forma di file XML_G (con estensione .scrpl).

3.12.4.4.13 addCharacter(Character character)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - character: il personaggio da aggiungere allo sceneggiato.
- **Descrizione:** Aggiunge il Character passato per parametro allo Screenplay.

3.12.4.4.14 importCharacters(String screenplay, Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - screenplay: lo screenplay da cui importare i personaggi;
 - context: il contesto della view.
- **Descrizione:** Importa i personaggi da uno Screenplay, facendo copie profonde.

3.12.4.4.15 goToListSpeechesActivity(Context context, String selected)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view;
 - selected: il titolo del capitolo da aprire.
- **Descrizione:** Istanzia un ChapterPresenter passando per parametro un Chapter, identificato attraverso il nome indicato dal parametro selected. Successivamente assegna al campo statico di ListSpeechesActivity questo presenter e, attraverso una Intent, porta l'utente in questa Activity.

3.12.4.4.16 goToListCharactersActivity(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Istanzia un CharacterPresenter. Successivamente assegna al campo statico di ListCharactersActivity questo presenter e, attraverso una Intent, porta l'utente in questa Activity.

3.12.4.4.17 goToEditChapterActivity(Context context, String selected)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view;
 - selected: il nome che identifica il capitolo da modificare.
- **Descrizione:** Istanzia un ChapterPresenter passando per parametro un Chapter, identificato attraverso il nome indicato dal parametro selected. Successivamente assegna al campo statico di ListSpeechesActivity questo presenter e, attraverso una Intent, porta l'utente in questa Activity.

3.12.4.4.18 goToNewChapterActivity(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Istanzia un ChapterPresenter. Successivamente assegna al campo statico di NewChapterActivity questo presenter e, attraverso una Intent, porta l'utente in questa Activity.

3.12.4.4.19 goToNewCharacterActivity(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Istanzia un CharacterPresenter. Successivamente assegna al campo statico di NewCharacterActivity questo presenter e, attraverso una Intent, porta l'utente in questa Activity.

3.12.5 ScreenplayPresenterImpl (class)

ScreenplayPresenterImpl
<pre> - newScreenplayInterface : NewScreenplayInterface - newChapterInterface : NewChapterInterface - screenplay : Screenplay - listChapterInterface : ListChapterInterface - titlesAdapter : ArrayAdapter<String> + setActivity(listChapterInterface : List) : void + setActivity(newChapterInterface : NewChapterInterface) : void + getTitlesAdapter(context : Context, screenplay : String) : ArrayAdapter<String> + getScreenplayTitle() : String + getScreenplay() : Screenplay + export() : void + share() : void + newChapter(title : String) : void + newScreenplay(title : String, context : Context) : void + moveDownChapter(chapter : String) : void + moveUpChapter(chapter : String) : void + save(screenplay : Screenplay, context : Context) : boolean + addCharacter(character : Character) : void + importCharacter(screenplay : String, context : Context) : void + goToListSpeechesActivity(context : Context, selected : String) : void + goToListCharactersActivity(context : Context) : void + goToEditChapterActivity(context : Context, selected : String) : void + goToNewChapterActivity(context : Context) : void + goToNewCharacterActivity(context : Context) : void + loadChapterTitles(screenplayTitle : String, context : Context) : Vector<String> + ScreenplayPresenterImpl(listChapterActivity : ListChapterInterface) + ScreenplayPresenterImpl(newScreenplayInterface : NewScreenplayInterface) + ScreenplayPresenterImpl(newChapterInterface : NewChapterInterface) + ScreenplayPresenterImpl(listChapterActivity : ListChapterInterface) </pre>

Figura 49: Classe Sivodim::Drama::Presenter::ScreenplayPresenterImpl

3.12.5.1 Descrizione

Classe che implementa l'interfaccia ScreenplayPresenter.

3.12.5.2 Utilizzo

Viene usata per implementare il contratto del *presenter* associato alle ListChapterActivity, ShareActivity, ExportActivity e NewScreenplayActivity presenti nella componente Sivodim::Drama::View.

3.12.5.3 Classi ereditate

- Sivodim::Drama::Presenter::ScreenplayPresenter.

3.12.5.4 Attributi

3.12.5.4.1 newScreenplayInterface

- **Accesso:** private;
- **Tipo:** NewScreenplayInterface;

- **Descrizione:** Riferimento alla view NewScreenplayInterface che ScreenplayPresenterImpl utilizza per interagire con l'interfaccia utente, nel rispetto del pattern MVP.

3.12.5.4.2 newChapterInterface

- **Accesso:** private;
- **Tipo:** NewChapterInterface;
- **Descrizione:** Riferimento alla view NewChapterInterface che ScreenplayPresenterImpl utilizza per interagire con l'interfaccia utente, nel rispetto del pattern MVP.

3.12.5.4.3 screenplay

- **Accesso:** private;
- **Tipo:** Screenplay;
- **Descrizione:** Riferimento a uno sceneggiato di tipo Screenplay. I dati del Model vengono passati al Presenter per interagire in modo controllato con la relativa Activity, nel rispetto del pattern MVP.

3.12.5.4.4 listChapterInterface

- **Accesso:** private;
- **Tipo:** ListChapterInterface;
- **Descrizione:** Riferimento alla view ListChapterInterface che ScreenplayPresenterImpl utilizza per interagire con l'interfaccia utente, nel rispetto del pattern MVP.

3.12.5.4.5 titlesAdapter

- **Accesso:** private;
- **Tipo:** ArrayAdapter<String>;
- **Descrizione:** ArrayAdapter contenente la lista di titoli relativi a tutti gli sceneggiati presenti nella memoria SD del dispositivo.

3.12.5.5 Metodi

3.12.5.5.1 ScreenplayPresenterImpl(Screenplay screenplay)

- **Accesso:** public;
- **Argomenti:**
 - screenplay: lo Screenplay con cui inizializzare il presenter.
- **Descrizione:** Costruttore in quattro versioni: può ricevere per parametro un riferimento a Screenplay, NewScreenplayInterface, NewChapterInterface o ListChapterInterface.

3.12.5.5.2 setActivity(ListChapterInterface listChapterInterface)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - listChapterInterface: l'activity da registrare nel presenter.
- **Descrizione:** Metodo sovraccaricato disponibile in due versioni, che implementano i metodi definiti nell'interfaccia.

3.12.5.5.3 getScreenplay()

- **Accesso:** public;
- **Tipo di ritorno:** Screenplay;
- **Descrizione:** Implementa il metodo definito nell'interfaccia ScreenplayPresenter restituendo this.screenplay.

3.12.5.5.4 getScreenplayTitle()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Implementa il metodo definito nell'interfaccia ScreenplayPresenter utilizzando il metodo getTitle di Screenplay.

3.12.5.5.5 getTitlesAdapter(Context context, String screenplay)

- **Accesso:** public;
- **Tipo di ritorno:** ArrayAdapter<String>;
- **Argomenti:**

- context: il contesto della view;
- screenplay: lo sceneggiato da aprire.
- **Descrizione:** Implementa il metodo definito nell'interfaccia ScreenplayPresenter re-stituendo un ArrayAdapter di String con i titoli dei capitoli dello Screenplay.

3.12.5.5.6 goToListSpeechesActivity(Context context, String selected)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view;
 - selected: il capitolo selezionato.
- **Descrizione:** Istanzia un ChapterPresenter passando per parametro un Chapter, identificato attraverso il nome indicato dal parametro selected. Successivamente assegna al campo statico di ListSpeechesActivity questo presenter e, attraverso una Intent, porta l'utente in questa Activity.

3.12.5.5.7 goToListCharactersActivity(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Istanzia un CharacterPresenter. Successivamente assegna al campo statico di ListCharactersActivity questo presenter e, attraverso una Intent, porta l'utente in questa Activity.

3.12.5.5.8 goToEditChapterActivity(Context context, String selected)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view;
 - selected: il nome che identifica il capitolo.
- **Descrizione:** Istanzia un ChapterPresenter passando per parametro un Chapter, identificato attraverso il nome indicato dal parametro selected. Successivamente assegna al campo statico di ListSpeechesActivity questo presenter e, attraverso una Intent, porta l'utente in questa Activity.

3.12.5.5.9 goToNewChapterActivity(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Istanzia un ChapterPresenter. Successivamente assegna al campo statico di NewChapterActivity questo presenter e, attraverso una Intent, porta l'utente in questa Activity.

3.12.5.5.10 goToNewCharacterActivity(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Istanzia un CharacterPresenter. Successivamente assegna al campo statico di NewCharacterActivity questo presenter e, attraverso una Intent, porta l'utente in questa Activity.

3.12.5.5.11 export()

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Descrizione:** Implementa il metodo definito nell'interfaccia ScreenplayPresenter utilizzando le funzionalità di esportazione dell'attributo Screenplay.

3.12.5.5.12 share()

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Descrizione:** Implementa il metodo definito nell'interfaccia ScreenplayPresenter utilizzando le funzionalità di condivisione dell'attributo Screenplay.

3.12.5.5.13 newScreenplay(String title, Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - title: il titolo dello sceneggiato;
 - context: il contesto della view.
- **Descrizione:** Implementa il metodo definito nell'interfaccia ScreenplayPresenter creando un nuovo Screenplay e salvandolo in memoria.

3.12.5.5.14 addCharacter(Character character)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - character: il personaggio da aggiungere allo sceneggiato.
- **Descrizione:** Implementa il metodo definito nell'interfaccia ScreenplayPresenter aggiungendo allo Screenplay il Character passato per riferimento.

3.12.5.5.15 importCharacters(String screenplay, Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - screenplay: il titolo che identifica lo sceneggiato da cui importare i personaggi;
 - context: il contesto della view.
- **Descrizione:** Implementa il metodo definito nell'interfaccia ScreenplayPresenter utilizzando i metodi di copia profonda dei Characters per importare i personaggi di uno sceneggiato in quello corrente.

3.12.5.5.16 newChapter(String title)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - title: il titolo del capitolo da creare.
- **Descrizione:** Implementa il metodo definito nell'interfaccia ScreenplayPresenter creando un nuovo capitolo e aggiungendolo allo sceneggiato.

3.12.5.5.17 moveUpChapter(String chapterTitle)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - chapterTitle: il titolo del capitolo da spostare.
- **Descrizione:** Implementa il metodo definito nell'interfaccia ScreenplayPresenter estraendo il Chapter dal contenitore e reinserendolo nella posizione desiderata.

3.12.5.5.18 moveDownChapter(String chapterTitle)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - chapterTitle: il titolo del capitolo da spostare.
- **Descrizione:** Implementa il metodo definito nell'interfaccia ScreenplayPresenter estraendo il Chapter dal contenitore e reinserendolo nella posizione desiderata.

3.12.5.5.19 save(Screenplay screenplay, Context context)

- **Accesso:** public;
- **Tipo di ritorno:** boolean;
- **Argomenti:**
 - screenplay: lo Screenplay da salvare;
 - context: il contesto della view.
- **Descrizione:** Implementa il metodo definito nell'interfaccia ScreenplayPresenter salvando in memoria lo sceneggiato nella cartella di default dell'app.

3.12.6 ChapterPresenter (interface)

ChapterPresenterImpl
<pre> - chapter : Chapter - characterContainer : CharacterContainer - projectName : String - speechSelected : int - speechArrayAdapter : SpeechArrayAdapter - listSpeechesInterface : ListSpeechesInterface - newChapterInterface : NewChapterInterface - editChapterInterface : EditChapterInterface - newSpeechInterface : NewSpeechInterface + orderSpeech() : void + loadSpeeches(context : Context) : void + setActivity(listSpeechesInterface : ListSpeechesInterface) : void + setActivity(editChapterInterface : EditChapterInterface) : void + setActivity(newSpeechInterface : NewSpeechInterface) : void + getSpeeches(context : Context) : SpeechArrayAdapter + getCharactersAdapter(context : Context) : ArrayAdapter<String> + getChapterTitle() : String + getSpeechSelected() : int + setChapterTitle(title : String) : void + setSpeechSelected(index : int) : void + newSpeech(text : String, characterName : String, emotion : String) : void + deleteSpeech(speech : Speech) : void + goToEditSpeechActivity(context : Context, selected : Speech) : void + goToListCharactersActivity(context : Context) : void + goUpEditChapterActivity(context : Context) : void + goToNewCharacterActivity(context : Context) : void + goToNewSpeechActivity(context : Context) : void + ChapterPresenterImpl(chapter : Chapter, characterContainer : CharacterContainer, projectName : String) + ChapterPresenterImpl(listSpeechesInterface : ListSpeechesInterface) + ChapterPresenterImpl(editChapterInterface : EditChapterInterface) + ChapterPresenterImpl(newChapterInterface : NewChapterInterface) </pre>

Figura 50: Classe Sivodim::Drama::Presenter::ChapterPresenter

3.12.6.1 Descrizione

Interfaccia che rappresenta il contratto attraverso cui implementare il *presenter* che gestisce le attività relative ai capitoli di uno sceneggiato.

3.12.6.2 Utilizzo

Viene usata per definire il contratto di implementazione del *presenter* associato alle ListSpeechesActivity, NewChapterActivity e EditChapterActivity presenti nel package Sivodim::Drama::View.

3.12.6.3 Classi figlie

- Sivodim::Drama::Presenter::ChapterPresenterImpl.

3.12.6.4 Metodi

3.12.6.4.1 moveUpSpeech(Speech speech)

- **Accesso:** public;
- **Tipo di ritorno:** void;

- **Argomenti:**
 - speech: lo Speech da spostare.
- **Descrizione:** Sposta indietro di una posizione una battuta del capitolo (rispetto l'ordine delle battute).

3.12.6.4.2 moveDownSpeech(Speech speech)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - speech: lo Speech da spostare.
- **Descrizione:** Sposta avanti di una posizione una battuta del capitolo (rispetto l'ordine delle battute).

3.12.6.4.3 setActivity(NewSpeechInterface newSpeechInterface)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - newSpeechInterface: l'activity che utilizza il presenter.
- **Descrizione:** Metodo sovraccaricato che permette alle *activities* che fanno uso di questo presenter di registrarsi, in accordo al pattern MVP.

3.12.6.4.4 getSpeeches(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** SpeechArrayAdapter;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Restituisce uno SpeechArrayAdapter contenente le battute del capitolo.

3.12.6.4.5 getCharactersAdapter(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** ArrayAdapter<String>;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Restituisce un ArrayAdapter di String contenente i nomi dei personaggi dello sceneggiato.

3.12.6.4.6 getChapterTitle()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Restituisce il titolo del capitolo.

3.12.6.4.7 getSpeechSelected()

- **Accesso:** public;
- **Tipo di ritorno:** int;
- **Descrizione:** Restituisce l'ultimo Speech selezionato nella ListSpeechesActivity.

3.12.6.4.8 setChapterTitle(String title)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - title: il titolo da dare al capitolo.
- **Descrizione:** Imposta il titolo del capitolo in base alla String passata per parametro.

3.12.6.4.9 setSpeechSelected(int position)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - position: l'indice della battuta.
- **Descrizione:** Memorizza la selezione di una Speech tenendo traccia del suo indice all'interno del capitolo.

3.12.6.4.10 newSpeech(String text, String characterName, String emotion)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - text: il testo della battuta;
 - characterName: il nome del personaggio;
 - emotion: il nome che identifica l'emozione.
- **Descrizione:** Crea una nuova battuta sulla base dei parametri ricevuti e la aggiunge al capitolo.

3.12.6.4.11 deleteSpeech(Speech speech)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - speech: la Speech da togliere.
- **Descrizione:** Elimina dal capitolo la battuta il cui riferimento è passato per parametro.

3.12.6.4.12 goToEditSpeechActivity(Context context, Speech selected)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view;
 - selected: la Speech selezionata.
- **Descrizione:** Istanzia uno SpeechPresenter. Successivamente assegna al campo statico di EditSpeechActivity questo presenter e, attraverso una Intent, porta l'utente in questa Activity.

3.12.6.4.13 goToListCharactersActivity(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Istanzia un CharacterPresenter. Successivamente assegna al campo statico di ListCharactersActivity questo presenter e, attraverso una Intent, porta l'utente in questa Activity.

3.12.6.4.14 goToEditChapterActivity(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Istanzia un ChapterPresenter. Successivamente assegna al campo statico di EditChapterActivity questo presenter e, attraverso una Intent, porta l'utente in questa Activity.

3.12.6.4.15 goToNewCharacterActivity(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Istanzia un CharacterPresenter. Successivamente assegna al campo statico di NewCharacterActivity questo presenter e, attraverso una Intent, porta l'utente in questa Activity.

3.12.6.4.16 goToNewSpeechActivity(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Istanzia uno SpeechPresenter. Successivamente assegna al campo statico di NewSpeechActivity questo presenter e, attraverso una Intent, porta l'utente in questa Activity.

3.12.7 ChapterPresenterImpl (class)

ChapterPresenterImpl
<pre> - chapter : Chapter - characterContainer : CharacterContainer - projectName : String - speechSelected : int - speechArrayAdapter : SpeechArrayAdapter - listSpeechesInterface : ListSpeechInterface - newChapterInterface : NewChapterInterface - editChapterInterface : EditChapterInterface - newSpeechInterface : NewSpeechInterface + orderSpeech() : void + loadSpeeches(context : Context) : void + setActivity(listSpeechesInterface : ListSpeechesInterface) : void + setActivity(editChapterInterface : EditChapterInterface) : void + setActivity(newSpeechInterface : NewSpeechInterface) : void + getSpeeches(context : Context) : SpeechArrayAdapter + getCharactersAdapter(context : Context) : ArrayAdapter<String> + getChapterTitle() : String + getSpeechSelected() : int + setChapterTitle(title : String) : void + setSpeechSelected(index : int) : void + newSpeech(text : String, characterName : String, emotion : String) : void + deleteSpeech(speech : Speech) : void + goToEditSpeechActivity(context : Context, selected : Speech) : void + goToListCharactersActivity(context : Context) : void + goUpEditChapterActivity(context : Context) : void + goToNewCharacterActivity(context : Context) : void + goToNewSpeechActivity(context : Context) : void + ChapterPresenterImpl(chapter : Chapter, characterContainer : CharacterContainer, projectName : String) + ChapterPresenterImpl(listSpeechesInterface : ListSpeechesInterface) + ChapterPresenterImpl(editChapterInterface : EditChapterInterface) + ChapterPresenterImpl(newChapterInterface : NewChapterInterface) </pre>

Figura 51: Classe Sivodim::Drama::Presenter::ChapterPresenterImpl

3.12.7.1 Descrizione

Classe che implementa l'interfaccia ChapterPresenter.

3.12.7.2 Utilizzo

Viene usata per implementare il contratto del *presenter* associato alle ListSpeechesActivity, NewChapterActivity e EditChapterActivity presenti nel package_G Sivodim::Drama::View.

3.12.7.3 Classi ereditate

- Sivodim::Drama::Presenter::ChapterPresenter.

3.12.7.4 Attributi

3.12.7.4.1 chapter

- **Accesso:** private;
- **Tipo:** Chapter;

- **Descrizione:** Riferimento a un capitolo di tipo Chapter. I dati del Model vengono passati al Presenter per interagire in modo controllato con la relativa Activity, nel rispetto del pattern MVP.

3.12.7.4.2 characterContainer

- **Accesso:** private;
- **Tipo:** CharacterContainer;
- **Descrizione:** Riferimento alla lista di personaggi di tipo Character relativa allo sceneggiato aperto. I dati del Model vengono passati al Presenter per interagire in modo controllato con la relativa Activity, nel rispetto del pattern MVP.

3.12.7.4.3 projectName

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Nome dello sceneggiato all'interno del quale è contenuto il capitolo aperto.

3.12.7.4.4 speechSelected

- **Accesso:** private;
- **Tipo:** int;
- **Descrizione:** Intero utile a tenere traccia della selezione del capitolo.

3.12.7.4.5 speechArrayAdapter

- **Accesso:** private;
- **Tipo:** SpeechArrayAdapter;
- **Descrizione:** ArrayAdapter *custom* contenente la lista dinamica delle battute presenti all'interno del capitolo.

3.12.7.4.6 listSpeechesInterface

- **Accesso:** private;
- **Tipo:** ListSpeechesInterface;
- **Descrizione:** Riferimento alla view ListSpeechesInterface che ChapterPresenterImpl utilizza per interagire con l'interfaccia utente, nel rispetto del pattern MVP.

3.12.7.4.7 newChapterInterface

- **Accesso:** private;
- **Tipo:** NewChapterInterface;
- **Descrizione:** Riferimento alla view NewChapterInterface che ChapterPresenterImpl utilizza per interagire con l'interfaccia utente, nel rispetto del pattern MVP.

3.12.7.4.8 editChapterInterface

- **Accesso:** private;
- **Tipo:** EditChapterInterface;
- **Descrizione:** Riferimento alla view EditChapterInterface che ChapterPresenterImpl utilizza per interagire con l'interfaccia utente, nel rispetto del pattern MVP.

3.12.7.4.9 newSpeechInterface

- **Accesso:** private;
- **Tipo:** NewSpeechInterface;
- **Descrizione:** Riferimento alla view NewSpeechInterface che ChapterPresenterImpl utilizza per interagire con l'interfaccia utente, nel rispetto del pattern MVP.

3.12.7.5 Metodi

3.12.7.5.1 ChapterPresenterImpl(Chapter chapter, CharacterContainer characterContainer, String projectName)

- **Accesso:** public;
- **Argomenti:**
 - chapter: il capitolo da gestire;
 - characterContainer: il CharacterContainer dello Screenplay;
 - projectName: il titolo dello Screenplay.
- **Descrizione:** Il costruttore si presenta in quattro alternative:
 - ChapterPresenterImpl(Chapter chapter, CharacterContainer characterContainer, String projectName);
 - ChapterPresenterImpl(EditChapterInterface editChapterInterface);
 - ChapterPresenterImpl(ListSpeechesInterface listSpeechesInterface);
 - ChapterPresenterImpl(NewChapterInterface newChapterInterface).

3.12.7.5.2 setActivity(ListSpeechesInterface listSpeechesInterface)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - listSpeechesInterface: l'activity che deve registrarsi nel presenter.
- **Descrizione:** Metodo sovraccaricato che permette alle *activities* che fanno uso di questo presenter di registrarsene, in accordo al pattern MVP.

3.12.7.5.3 getChapterTitle()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Implementa il metodo definito nell'interfaccia ScreenplayPresenter restituendo il nome del capitolo attraverso l'attributo chapter.

3.12.7.5.4 getSpeeches()

- **Accesso:** public;
- **Tipo di ritorno:** SpeechArrayAdapter;
- **Descrizione:** Implementa il metodo definito nell'interfaccia ScreenplayPresenter creando e restituendo uno SpeechArrayAdapter.

3.12.7.5.5 getCharactersAdapter(Contect context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Implementa il metodo definito nell'interfaccia ScreenplayPresenter restituendo un ArrayAdapter di String con i nomi dei personaggi dello sceneggiato, ottenuti attraverso le funzionalità del CharacterContainer.

3.12.7.5.6 getSpeechSelected()

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Descrizione:** Implementa il metodo definito nell'interfaccia ScreenplayPresenter restituendo la posizione che si era memorizzato con l'ultima selezione di Speech.

3.12.7.5.7 setChapterTitle(String title)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - title: il titolo da assegnare.
- **Descrizione:** Imposta il titolo sul Chapter che ha per attributo.

3.12.7.5.8 setSpeechSelected(int position)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - position: l'indice della battuta.
- **Descrizione:** Setta una variabile che mantiene come stato la posizione indicata nel parametro del metodo.

3.12.7.5.9 newSpeech(String text, String characterName, String emotion)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - text: il testo della battuta;
 - characterName: il nome del personaggio;
 - emotion: il nome che identifica l'emozione.
- **Descrizione:** Crea uno Speech e lo aggiunge al capitolo che gestisce.

3.12.7.5.10 deleteSpeech(Speech speech)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - speech: la Speech da togliere.
- **Descrizione:** Implementa il metodo definito nell'interfaccia ChapterPresenter eliminando dal capitolo la Speech passata per riferimento.

3.12.7.5.11 goToEditSpeechActivity(Context context, Speech selected)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view;
 - selected: la battuta da modificare.
- **Descrizione:** Istanzia uno SpeechPresenter. Successivamente assegna al campo statico di EditSpeechActivity questo presenter e, attraverso una Intent, porta l'utente in questa Activity.

3.12.7.5.12 goToListCharactersActivity(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Istanzia un CharacterPresenter. Successivamente assegna al campo statico di ListCharactersActivity questo presenter e, attraverso una Intent, porta l'utente in questa Activity.

3.12.7.5.13 goToEditChapterActivity(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Istanzia un CharacterPresenter. Successivamente assegna al campo statico di EditChapterActivity questo presenter e, attraverso una Intent, porta l'utente in questa Activity.

3.12.7.5.14 goToNewCharacterActivity(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view.

- **Descrizione:** Istanzia un CharacterPresenter. Successivamente assegna al campo statico di NewCharacterActivity questo presenter e, attraverso una Intent, porta l'utente in questa Activity.

3.12.7.5.15 goToNewSpeechActivity(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Istanzia uno SpeechPresenter. Successivamente assegna al campo statico di NewSpeechActivity questo presenter e, attraverso una Intent, porta l'utente in questa Activity.

3.12.7.5.16 moveUpSpeech(Speech speech)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - speech: la battuta da spostare.
- **Descrizione:** Estrae dal contenitore di Speech lo Speech selezionato e lo reinserisce nella posizione desiderata.

3.12.7.5.17 moveDownSpeech(Speech speech)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - speech: la battuta da spostare.
- **Descrizione:** Estrae dal contenitore di Speech lo Speech selezionato e lo reinserisce nella posizione desiderata.

3.12.8 SpeechPresenter (interface)

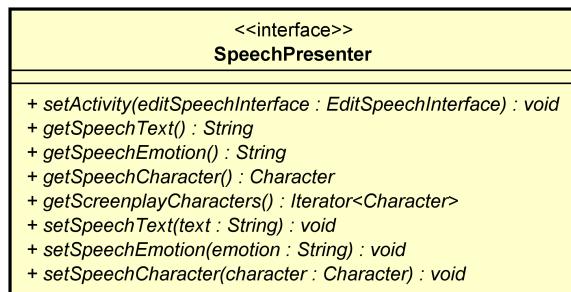


Figura 52: Classe Sivodim::Drama::Presenter::SpeechPresenter

3.12.8.1 Descrizione

Interfaccia che rappresenta il contratto attraverso cui implementare il *presenter* delle pagine relative alla gestione delle battuta.

3.12.8.2 Utilizzo

Viene usata per definire il contratto di implementazione del *presenter* associato alle New-SpeechActivity ed EditSpeechActivity presenti nella componente Sivodim::Drama::View.

3.12.8.3 Classi figlie

- Sivodim::Drama::Presenter::SpeechPresenterImpl.

3.12.8.4 Metodi

3.12.8.4.1 setActivity(EditSpeechInterface editSpeechInterface)

- Accesso:** public;
- Tipo di ritorno:** void;
- Argomenti:**
 - editSpeechInterface: EditSpeechInterface che si intende registrare al presenter.
- Descrizione:** Registra una EditSpeechInterface al presenter, in accordo al modello MVP.

3.12.8.4.2 getSpeechText()

- Accesso:** public;
- Tipo di ritorno:** String;
- Descrizione:** Restituisce il testo della battuta presente nello stato.

3.12.8.4.3 getSpeechEmotion()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Restituisce la String che identifica l'emozione associata alla battuta presente nello stato.

3.12.8.4.4 getSpeechCharacter()

- **Accesso:** public;
- **Tipo di ritorno:** Character;
- **Descrizione:** Restituisce il personaggio associato alla battuta presente nello stato del presenter.

3.12.8.4.5 getScreenplayCharacters()

- **Accesso:** public;
- **Tipo di ritorno:** Iterator<Character>;
- **Descrizione:** Restituisce un iteratore sui personaggi dello sceneggiato utilizzando il characterContainer presente nello stato.

3.12.8.4.6 setSpeechText(String text)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - text: String con il testo da dare alla battuta.
- **Descrizione:** Assegna alla battuta presente nello stato del presenter il testo passato per parametro.

3.12.8.4.7 setSpeechEmotion(String emotion)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - emotion: String che identifica l'emozione da associare alla battuta.
- **Descrizione:** Assegna l'emozione specificata alla battuta presenter nello stato del presenter.

3.12.8.4.8 setSpeechCharacter(Character character)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - character: il personaggio da assegnare alla battuta.
- **Descrizione:** Assegna il personaggio specificato alla battuta presente nello stato del presenter.

3.12.9 SpeechPresenterImpl (class)

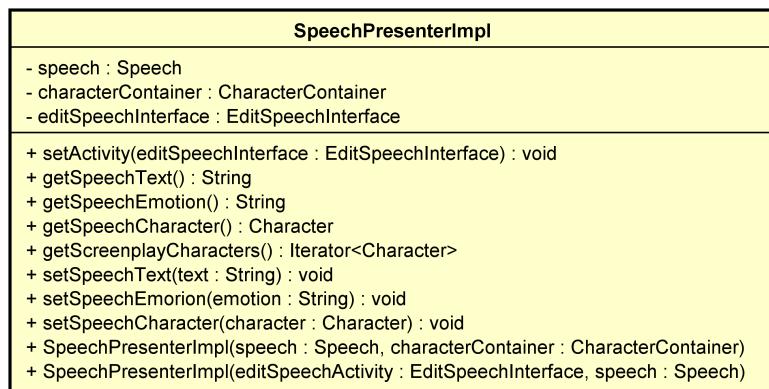


Figura 53: Classe Sivodim::Drama::Presenter::SpeechPresenterImpl

3.12.9.1 Descrizione

Classe che implementa l'interfaccia SpeechPresenter.

3.12.9.2 Utilizzo

Viene usata per implementare il contratto del *presenter* associato alle NewSpeechActivity ed EditSpeechActivity presenti nella componente Sivodim::Drama::View.

3.12.9.3 Classi ereditate

- Sivodim::Drama::Presenter::SpeechPresenter.

3.12.9.4 Attributi

3.12.9.4.1 characterContainer

- **Accesso:** private;
- **Tipo:** CharacterContainer;

- **Descrizione:** Riferimento al contenitore di personaggi relativi a uno specifico sceneggiato, necessario per associare alla battuta un certo personaggio della lista.

3.12.9.4.2 editSpeechInterface

- **Accesso:** private;
- **Tipo:** EditSpeechInterface;
- **Descrizione:** Riferimento alla view EditSpeechInterface che SpeechPresenterImpl utilizza per interagire con l'interfaccia utente, nel rispetto del pattern MVP.

3.12.9.4.3 speech

- **Accesso:** private;
- **Tipo:** Speech;
- **Descrizione:** Riferimento a una battuta di tipo Speech. I dati del Model vengono passati al Presenter per interagire in modo controllato con la relativa Activity, nel rispetto del pattern MVP.

3.12.9.5 Metodi

3.12.9.5.1 SpeechPresenterImpl(Speech speech, CharacterContainer characterContainer)

- **Accesso:** public;
- **Argomenti:**
 - speech: lo Speech che viene gestito dal presenter;
 - characterContainer: il CharacterContainer dello sceneggiato cui la Speech appartiene.
- **Descrizione:** Costruttore che memorizza una Speech e un CharacterContainer.

3.12.9.5.2 SpeechPresenterImpl(EditSpeechInterface editSpeechActivity, Speech speech)

- **Accesso:** public;
- **Argomenti:**
 - editSpeechInterface: interfaccia dell'activity che utilizza il presenter;
 - speech: la Speech che si intende gestire.
- **Descrizione:** Costruttore che prende la Speech da gestire e la activity che si occupa della sua gestione.

3.12.9.5.3 setActivity(EditSpeechInterface editSpeechInterface)

- **Accesso:** public;
- **Argomenti:**
 - editSpeechInterface: l'activity che utilizza il presenter
- **Descrizione:** Inizializza un riferimento all'activity che utilizza il presenter, in accordo al modello MVP.

3.12.9.5.4 getSpeechText()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Restituisce speech.getText().

3.12.9.5.5 getSpeechEmotion()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Restituisce speech.getEmotion().

3.12.9.5.6 getScreenplayCharacters()

- **Accesso:** public;
- **Tipo di ritorno:** Iterator<Character>;
- **Descrizione:** Restituisce characterContainer.iterator().

3.12.9.5.7 getSpeechCharacter()

- **Accesso:** public;
- **Tipo di ritorno:** Character;
- **Descrizione:** Restituisce speech.getCharacter().

3.12.9.5.8 setSpeechText(String text)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - text: il testo della battuta.
- **Descrizione:** Assegna il testo alla battuta attraverso il metodo setText di Speech.

3.12.9.5.9 setSpeechEmotion(String emotion)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - emotion: l'emozione da dare alla battuta.
- **Descrizione:** Assegna un'emozione alla battuta attraverso il metodo setEmotion di Speech.

3.12.9.5.10 setSpeechCharacter(Character character)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - character: il personaggio da dare alla battuta.
- **Descrizione:** Assegna un personaggio alla battuta attraverso il metodo setCharacter di Speech.

3.12.10 CharacterPresenter (interface)

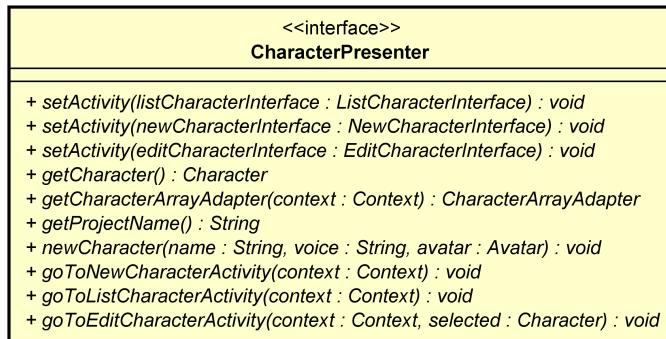


Figura 54: Classe Sivodim::Drama::Presenter::CharacterPresenter

3.12.10.1 Descrizione

Interfaccia che rappresenta il contratto attraverso cui implementare il *presenter* delle pagine relative alla gestione dei personaggi.

3.12.10.2 Utilizzo

Viene usata per definire il contratto di implementazione del *presenter* associato alle NewCharacterActivity, ListCharacterActivity ed EditCharacterActivity presenti nella componente Sivodim::Drama::View.

3.12.10.3 Classi figlie

- Sivodim::Drama::Presenter::CharacterPresenterImpl.

3.12.10.4 Metodi

3.12.10.4.1 setActivity(ListCharacterInterface listCharacterInterface)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - listCharacterInterface: l'activity che utilizza il presenter.
- **Descrizione:** Registra nel presenter l'activity che lo utilizza, in accordo al modello MVP.

3.12.10.4.2 setActivity(NewCharacterInterface newCharacterInterface)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - newCharacterInterface: l'activity che utilizza il presenter.
- **Descrizione:** Registra nel presenter l'activity che lo utilizza, in accordo con MVP.

3.12.10.4.3 setActivity(EditCharacterInterface editCharacterInterface)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - editCharacterInterface: l'activity che utilizza il presenter.
- **Descrizione:** Registra nel presenter l'activity che lo utilizza, in accordo al modello MVP.

3.12.10.4.4 getCharacter()

- **Accesso:** public;
- **Tipo di ritorno:** Character;
- **Descrizione:** Restituisce un riferimento al personaggio che il presenter sta gestendo.

3.12.10.4.5 getCharacterAdapter(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** CharacterAdapter;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Costruisce e restituisce un CharacterAdapter con i personaggi dello sceneggiato.

3.12.10.4.6 getProjectName()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Restituisce il titolo dello Screenplay.

3.12.10.4.7 newCharacter(String name, String voice, Avatar avatar)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - name: il nome del personaggio;
 - voice: il nome che identifica una voce;
 - avatar: un riferimento ad Avatar.
- **Descrizione:** Crea un nuovo personaggio e lo aggiunge al CharacterContainer dello Screenplay.

3.12.10.4.8 goToEditCharacterActivity(Context context, Character character)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view;
 - character: un riferimento a Character.
- **Descrizione:** Porta l'utente a visualizzare la EditChatacerActivity.

3.12.10.4.9 goToNewCharacterActivity(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Sposta la view alla NewCharacterActivity.

3.12.10.4.10 goToListCharacterActivity(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Sposta la view alla ListChapterActivity.

3.12.11 CharacterPresenterImpl (class)

CharacterPresenterImpl	
<ul style="list-style-type: none"> - projectName : String - characterContainer : CharacterContainer - character : Character - characterInterface : NewCharacterInterface - listCharacterInterface : ListCharacterInterface - editCharacterInterface : EditCharacterInterface 	
<ul style="list-style-type: none"> + setActivity(listCharacterInterface : ListCharacterInterface) : void + setActivity(newCharacterInterface : NewCharacterInterface) : void + setActivity(editCharacterInterface : EditCharacterInterface) : void + getCharacter() : Character + getCharacterArrayAdapter(context : Context) : CharacterArrayAdapter + getProjectName() : String + newCharacter(name : String, voice : String, avatar : Avatar) : void + goToNewCharacterActivity(context : Context) : void + goToListCharacterActivity(context : Context) : void + goUpEditCharacterActivity(context : Context, selected : Character) : void + CharacterPresenterImpl(characterContainer : CharacterContainer, projectName : String) + CharacterPresenterImpl(character : Character, projectName : String) + CharacterPresenterImpl(characterInterface : NewCharacterInterface) + CharacterPresenterImpl(listCharacterInterface : ListCharacterInterface) 	

Figura 55: Classe Sivodim::Drama::Presenter::CharacterPresenterImpl

3.12.11.1 Descrizione

Classe che implementa l'interfaccia CharacterPresenter.

3.12.11.2 Utilizzo

Viene usata per implementare il contratto del *presenter* associato alle NewCharacterActivity, EditCharacterActivity e ListCharacterActivity presenti nella componente Sivodim::Drama::View.

3.12.11.3 Classi ereditate

- Sivodim::Drama::Presenter::CharacterPresenter.

3.12.11.4 Attributi

3.12.11.4.1 projectName

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Nome dello sceneggiato aperto.

3.12.11.4.2 characterContainer

- **Accesso:** private;
- **Tipo:** CharacterContainer;
- **Descrizione:** Riferimento alla lista di personaggi di tipo CharacterContainer relativa allo sceneggiato aperto. I dati del Model vengono passati al Presenter per interagire in modo controllato con la relativa Activity, nel rispetto del pattern MVP.

3.12.11.4.3 character

- **Accesso:** private;
- **Tipo:** Character;
- **Descrizione:** Riferimento a un personaggio di tipo Character. I dati del Model vengono passati al Presenter per interagire in modo controllato con la relativa Activity, nel rispetto del pattern MVP.

3.12.11.4.4 newCharacterInterface

- **Accesso:** private;
- **Tipo:** NewCharacterInterface;
- **Descrizione:** Riferimento alla view NewCharacterInterface che CharacterPresenterImpl utilizza per interagire con l'interfaccia utente, nel rispetto del pattern MVP.

3.12.11.4.5 listCharacterInterface

- **Accesso:** private;
- **Tipo:** ListCharacterInterface;
- **Descrizione:** Riferimento alla view ListCharacterInterface che CharacterPresenterImpl utilizza per interagire con l'interfaccia utente, nel rispetto del pattern MVP.

3.12.11.4.6 editCharacterInterface

- **Accesso:** private;
- **Tipo:** EditCharacterInterface;
- **Descrizione:** Riferimento alla view EditCharacterInterface che CharacterPresenterImpl utilizza per interagire con l'interfaccia utente, nel rispetto del pattern MVP.

3.12.11.5 Metodi

3.12.11.5.1 CharacterPresenterImpl(CharacterContainer characterContainer, String projectName)

- **Accesso:** public;
- **Argomenti:**
 - characterContainer: il CharacterContainer dello sceneggiato;
 - projectName: il nome dello Screenplay
- **Descrizione:** Costruttore che prende un riferimento al CharacterContainer e al nome dello sceneggiato.

3.12.11.5.2 CharacterPresenterImpl(Character character, String projectName)

- **Accesso:** public;
- **Argomenti:**
 - character: il personaggio che il presenter deve gestire;
 - projectName: il nome dello sceneggiato
- **Descrizione:** Costruttore che richiede il personaggio da gestire e il nome dello sceneggiato.

3.12.11.5.3 CharacterPresenterImpl(NewCharacterInterface characterInterface)

- **Accesso:** public;
- **Argomenti:**
 - characterInterface: l'activity che deve utilizzare il presenter.
- **Descrizione:** Costruttore che richiede l'activity che andrà ad utilizzare il presenter.

3.12.11.5.4 setActivity(ListCharacterInterface listCharacterInterface)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - listCharacterInterface: l'activity che si intende registrare nel presenter in accordo al modello MVP.
- **Descrizione:** Registra l'activity nel presenter in accordo al modello MVP.

3.12.11.5.5 setActivity(EditCharacterInterface editCharacterInterface)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - editCharacterInterface: l'activity che utilizzerà il presenter.
- **Descrizione:** Registra l'activity al presenter, in accordo col modello MVP.

3.12.11.5.6 setActivity(NewCharacterInterface newCharacterInterface)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - newCharacterInterface: l'activity che utilizzerà il presenter.
- **Descrizione:** Registra l'activity nel presenter, in accordo col pattern MVP.

3.12.11.5.7 getCharacterArrayAdapter(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** CharacterArrayAdapter;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Restituisce un CharacterArrayAdapter che costruisce, contenente i personaggi dello sceneggiato.

3.12.11.5.8 getCharacter()

- **Accesso:** public;
- **Tipo di ritorno:** Character;
- **Descrizione:** Restituisce un riferimento al personaggio gestito dal presenter.

3.12.11.5.9 getProjectName()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Restituisce il titolo dello sceneggiato.

3.12.11.5.10 newCharacter(String name, String voice, Avatar avatar)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - name: il nome del personaggio;
 - voice: la voce del personaggio;
 - avatar: riferimento ad Avatar da assegnare al personaggio.
- **Descrizione:** Crea un nuovo personaggio con i parametri indicati, e lo inserisce nel CharacterContainer dello sceneggiato.

3.12.11.5.11 goToEditCharacterActivity(Context context, Character selected)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view;
 - selected; il personaggio selezionato, da gestire.
- **Descrizione:** Porta alla visualizzazione dell>EditCharacterActivity.

3.12.11.5.12 goToNewCharacterActivity(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view
- **Descrizione:** Porta alla visualizzazione della NewCharacterActivity.

3.12.11.5.13 goToListCharacterActivity(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Porta alla visualizzazione della ListCharacterActivity.

3.12.12 CharacterArrayAdapter (class)

CharArrayAdapter	
- characterList : List<Character>	
- characterName : TextView	
- characterAvatar : ImageView	
+ CharacterArrayAdapter(context : Context, resource : int)	
+ add(object : Character) : void	
+ getCount() : int	
+ getItem(index : int) : Character	
+ getView(position : int, convertView : View, parent : ViewGroup) : View	

Figura 56: Classe Sivodim::Drama::Presenter::CharArrayAdapter

3.12.12.1 Descrizione

Ridefinizione di ArrayAdapter, che utilizza un layout personale con più widget.

3.12.12.2 Utilizzo

Viene utilizzato per mostrare nella stessa linea il nome di un personaggio e il suo avatar.

3.12.12.3 Attributi

3.12.12.3.1 characterList

- **Accesso:** private;
- **Tipo:** List<Character>;
- **Descrizione:** Mantiene la lista dei personaggi dello sceneggiato.

3.12.12.3.2 characterName

- **Accesso:** private;
- **Tipo:** TextView;
- **Descrizione:** Riferimento alla TextView in cui inserire il nome del personaggio.

3.12.12.3.3 characterAvatar

- **Accesso:** public;
- **Tipo:** ImageView;
- **Descrizione:** Riferimento alla ImageView in cui inserire l'avatar del personaggio.

3.12.12.4 Metodi

3.12.12.4.1 CharacterArrayAdapter(Context context, int resource)

- **Accesso:** public;
- **Argomenti:**
 - context: il contesto della view;
 - resource: il numero di elementi con cui inizializzare il contenitore.
- **Descrizione:** Costruttore che inizializza il contenitore di Character.

3.12.12.4.2 add(Character object)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - object: il Character da aggiungere al contenitore.
- **Descrizione:** Aggiunge al contenitore il Character ricevuto per parametro.

3.12.12.4.3 getCount()

- **Accesso:** public;
- **Tipo di ritorno:** int;
- **Descrizione:** Restituisce il numero di elementi contenuto nel contenitore.

3.12.12.4.4 getItem(int index)

- **Accesso:** public;
- **Tipo di ritorno:** Character;
- **Argomenti:**
 - index: la posizione all'interno del contenitore in cui è presente il Character cercato.
- **Descrizione:** Restituisce il Character presente nel contenitore alla posizione ricevuta per parametro.

3.12.12.4.5 getView(int position, View convertView, ViewGroup parent)

- **Accesso:** public;
- **Tipo di ritorno:** View;
- **Argomenti:**
 - position: la posizione del Character nel contenitore;
 - convertView: il layout in cui si vogliono inserire informazioni;
 - parent: il layout in cui si vuole inserire l'elemento.
- **Descrizione:** Costruisce e restituisce una view contenente l'elenco dei Character secondo il layout indicato.

3.12.13 SpeechArrayAdapter (class)

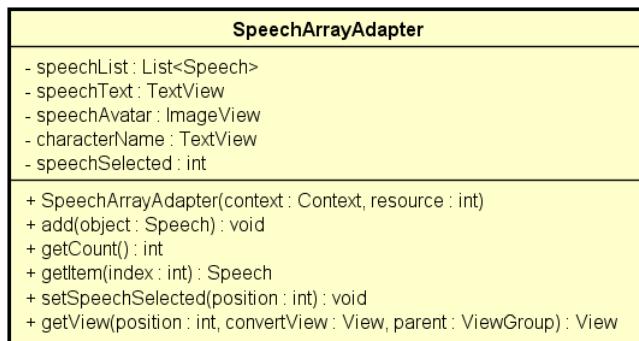


Figura 57: Classe Sivodim::Drama::Presenter::SpeechArrayAdapter

3.12.13.1 Descrizione

Ridefinizione di ArrayAdapter, che utilizza un layout personale con più widget.

3.12.13.2 Utilizzo

Viene utilizzato per mostrare nella stessa linea il testo di una battuta, il nome del personaggio associato e il suo avatar.

3.12.13.3 Attributi

3.12.13.3.1 speechList

- **Accesso:** private;
- **Tipo:** List<Speech>;
- **Descrizione:** Il contenitore che mantiene la lista delle battute da visualizzare.

3.12.13.3.2 speechText

- **Accesso:** private;
- **Tipo:** TextView;
- **Descrizione:** Riferimento alla TextView in cui inserire il testo della battuta.

3.12.13.3.3 speechAvatar

- **Accesso:** private;
- **Tipo:** ImageView;
- **Descrizione:** Riferimento alla ImageView in cui inserire l'avatar del personaggio associato alla battuta.

3.12.13.3.4 characterName

- **Accesso:** private;
- **Tipo:** TextView;
- **Descrizione:** Riferimento alla TextView in cui inserire il nome del personaggio associato alla battuta.

3.12.13.3.5 speechSelected

- **Accesso:** private;
- **Tipo:** int;
- **Descrizione:** Variabile di utilità in cui viene salvata la posizione della eventuale Speech selezionata.

3.12.13.4 Metodi

3.12.13.4.1 SpeechArrayAdapter(Context context, int resource)

- **Accesso:** public;
- **Argomenti:**
 - context: il contesto della view;
 - resource: il numero di elementi con cui inizializzare il contenitore di Speech.
- **Descrizione:** Costruttore che inizializza il contenitore di Speech.

3.12.13.4.2 add(Speech object)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - object: la Speech da aggiungere al contenitore.
- **Descrizione:** Aggiunge la Speech ricevuta per parametro al contenitore.

3.12.13.4.3 getCount()

- **Accesso:** public;
- **Tipo di ritorno:** int;
- **Descrizione:** Restituisce il numero di elementi presenti nel contenitore di Speech.

3.12.13.4.4 getItem(int index)

- **Accesso:** public;
- **Tipo di ritorno:** Speech;
- **Argomenti:**
 - index: l'indice in cui si trova la Speech cercata.
- **Descrizione:** Restituisce la Speech che si trova nella posizione indicata.

3.12.13.4.5 setSpeechSelected(int position)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - position: la posizione della Speech selezionata.
- **Descrizione:** Memorizza la posizione della Speech selezionata (l'intero ricevuto per parametro).

3.12.13.4.6 getView(int position, View convertView, ViewGroup parent)

- **Accesso:** public;
- **Tipo di ritorno:** View;
- **Argomenti:**
 - position: la posizione dell'elemento;
 - convertView: il layout in cui inserire le informazioni della Speech;
 - parent: il layout in cui inserire l'elemento creato.
- **Descrizione:** Costruisce e restituisce una view contenente l'elenco delle Speech secondo il layout indicato.

3.12.14 ArrayAdapter (class)

StringArrayAdapter
<ul style="list-style-type: none"> - textList>List<String> : int - stringselected : int - textView : TextView
<ul style="list-style-type: none"> + StringArrayAdapter(context : Context, resource : int) + add(object : String) : void + getCount() : int + getItem(index : int) : String + getView(position : int, convertView : View, parent : ViewGroup) : View + setStringSelected(position : int) : void

Figura 58: Classe Sivodim::Drama::Presenter::StringArrayAdapter

3.12.14.1 Descrizione

Ridefinizione di ArrayAdapter, necessario per reperire dinamicamente delle stringhe e rappresentarle visivamente attraverso una ListView nelle activity ListChapterActivity e HomeActivity.

3.12.14.2 Utilizzo

Viene utilizzato per visualizzare la lista di sceneggiati o capitoli (salvati) su schermo; permette inoltre di segnalare attraverso un colore la selezione di un elemento interno all'array.

3.12.14.3 Attributi

3.12.14.3.1 textList

- **Accesso:** private;
- **Tipo:** List<String>;
- **Descrizione:** Memorizza la lista di stringhe da visualizzare.

3.12.14.3.2 textView

- **Accesso:** private;
- **Tipo:** TextView;
- **Descrizione:** Riferimento alla TextView in cui viene visualizzato il testo.

3.12.14.3.3 stringSelected

- **Accesso:** private;
- **Tipo:** int;
- **Descrizione:** Memorizza la posizione dell'elemento stringa selezionato.

3.12.14.4 Metodi

3.12.14.4.1 StringArrayAdapter(Context context)

- **Accesso:** public;
- **Argomenti:**
 - context: il contesto della view;
- **Descrizione:** Costruttore che inizializza il contenitore di stringhe e il layout grafico.

3.12.14.4.2 add(String object)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - object: oggetto stringa da aggiungere all'attributo lista textList della classe.
- **Descrizione:** Aggiunge una nuova battuta da visualizzare su schermo.

3.12.14.4.3 getCount()

- **Accesso:** public;
- **Tipo di ritorno:** int;
- **Descrizione:** Restituisce il numero di stringhe contenute nella lista textList.

3.12.14.4.4 getItem(int index)

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Argomenti:**
 - index: la posizione all'interno del contenitore in cui è presente la stringa cercata.
- **Descrizione:** Restituisce la stringa presente nel contenitore alla posizione ricevuta per parametro.

3.12.14.4.5 setStringSelected(int position)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - position: intero che rappresenta la posizione dell'elemento selezionato.
- **Descrizione:** Assegna all'attributo stringSelected la posizione dell'elemento selezionato.

3.12.14.4.6 `getView(int position, View convertView, ViewGroup parent)`

- **Accesso:** public;
- **Tipo di ritorno:** View;
- **Argomenti:**
 - position: la posizione della stringa nel contenitore;
 - convertView: il layout in cui si vogliono inserire informazioni;
 - parent: il layout in cui si vuole inserire l'elemento.
- **Descrizione:** Costruisce e restituisce una view contenente l'elenco di stringhe secondo il layout indicato.

3.12.15 EmotionArrayAdapter (class)

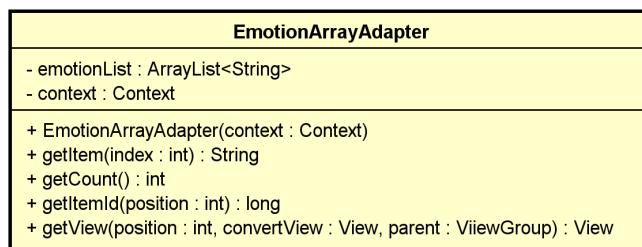


Figura 59: Classe Sivodim::Drama::Presenter::EmotionArrayAdapter

3.12.15.1 Descrizione

Ridefinizione di BaseAdapter, necessario per reperire dinamicamente le emozioni e rappresentarle visivamente attraverso una GridView in ListSpeechesActivity.

3.12.15.2 Utilizzo

Viene utilizzato per mostrare nella stessa "cella" il nome di un'emozione e la sua corrispettiva icona.

3.12.15.3 Attributi

3.12.15.3.1 emotionList

- **Accesso:** private;
- **Tipo:** ArrayList<String>;
- **Descrizione:** Memorizza la lista delle emozioni.

3.12.15.3.2 context

- **Accesso:** private;
- **Tipo:** Context;
- **Descrizione:** Riferimento al contesto dell'applicazione passato attraverso la View.

3.12.15.4 Metodi

3.12.15.4.1 EmotionArrayAdapter(Context context)

- **Accesso:** public;
- **Argomenti:**
 - context: il contesto della view;
- **Descrizione:** Costruttore che inizializza il contenitore di emozioni.

3.12.15.4.2 getItem(int index)

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Argomenti:**
 - index: la posizione all'interno del contenitore in cui è presente l'emozione cercata.
- **Descrizione:** Restituisce l'emozione presente nel contenitore alla posizione ricevuta per parametro.

3.12.15.4.3 getItemID(int position)

- **Accesso:** public;
- **Tipo di ritorno:** long;
- **Argomenti:**
 - position: intero che rappresenta la posizione.
- **Descrizione:** Metodo non utilizzato ma ridefinito per poter estendere correttamente la classe BaseAdapter.

3.12.15.4.4 `getView(int position, View convertView, ViewGroup parent)`

- **Accesso:** public;
- **Tipo di ritorno:** View;
- **Argomenti:**
 - position: la posizione dell'emozione nel contenitore;
 - convertView: il layout in cui si vogliono inserire informazioni;
 - parent: il layout in cui si vuole inserire l'elemento.
- **Descrizione:** Costruisce e restituisce una view contenente l'elenco di emozioni secondo il layout indicato.

3.13 Sivodim::Drama::View

3.13.1 Informazioni sul package

Di seguito viene illustrata, tramite un diagramma delle classi, la visione di insieme del package Drama::View contenente tutte le classi relative al View nel rispetto del pattern MVP.

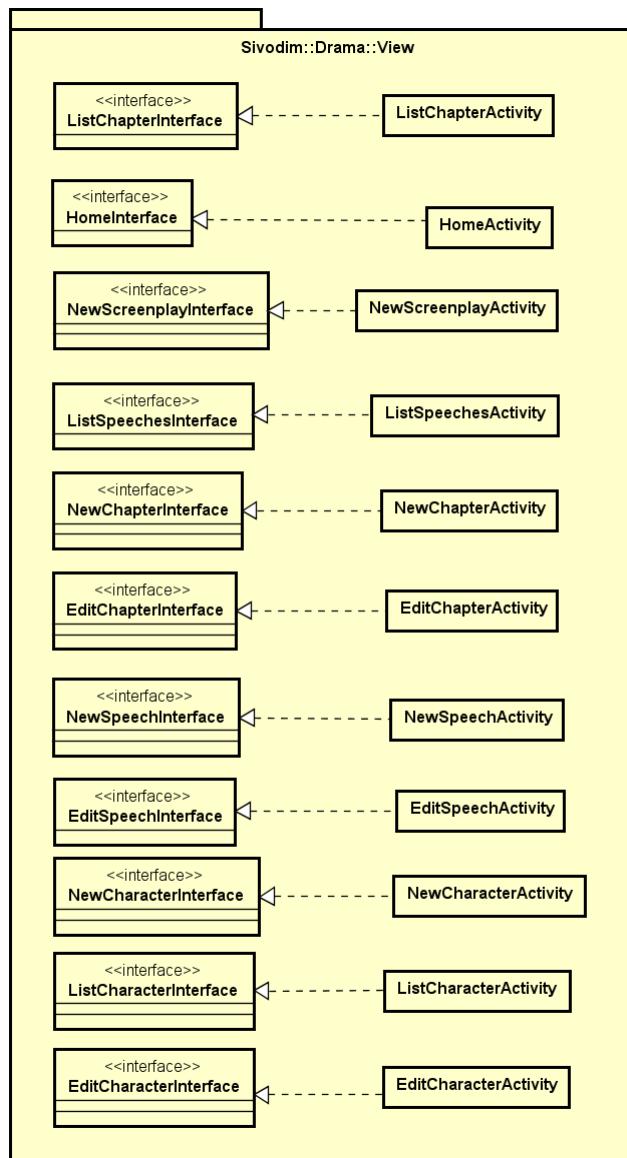


Figura 60: Componente Sivodim::Drama::View

3.13.2 HomeInterface (interface)



Figura 61: Classe Sivodim::Drama::View::HomeInterface

3.13.2.1 Descrizione

Interfaccia che rappresenta il contratto attraverso cui implementare la view della pagina principale dell'applicazione e catturarne gli eventi.

3.13.2.2 Utilizzo

Viene usata per definire il contratto di interazione con l'utente nella pagina principale dell'applicazione.

3.13.2.3 Classi figlie

- Sivodim::Drama::View::HomeActivity.

3.13.3 HomeActivity (class)

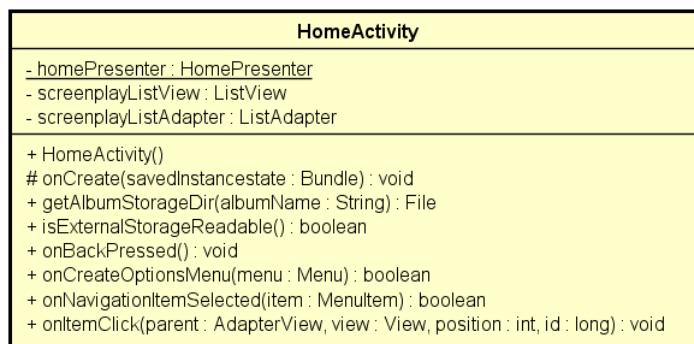


Figura 62: Classe Sivodim::Drama::View::HomeActivity

3.13.3.1 Descrizione

Classe che implementa l'interfaccia HomeActivity.

3.13.3.2 Utilizzo

Implementa il contratto di interazione con l'utente nella pagina principale dell'applicazione.

3.13.3.3 Classi ereditate

- Sivodim::Drama::View::HomeInterface.

3.13.3.4 Attributi

3.13.3.4.1 homePresenter

- **Accesso:** private;
- **Tipo:** HomePresenter;
- **Descrizione:** Riferimento statico al HomePresenter che la view HomeActivity utilizza per interagire con il model, nel rispetto del pattern MVP.

3.13.3.4.2 screenplayListView

- **Accesso:** private;
- **Tipo:** ListView;
- **Descrizione:** Componente grafica contenente la lista di sceneggiati da visualizzare nella *home page* dell'applicazione.

3.13.3.4.3 screenplayListAdapter

- **Accesso:** private;
- **Tipo:** ListAdapter;
- **Descrizione:** ListAdapter contenente la lista dinamica degli sceneggiati salvati nella memoria SD del telefono e a cui è associata la componente grafica screenplayListView.

3.13.3.5 Metodi

3.13.3.5.1 onCreate(Bundle savedInstanceState)

- **Accesso:** protected;
- **Tipo di ritorno:** void;
- **Descrizione:** Inizializza l'activity prelevando le informazioni necessarie dal presenter e predisponendo i listeners previsti.

3.13.4 ListChapterInterface (interface)



Figura 63: Classe Sivodim::Drama::View::ListChapterInterface

3.13.4.1 Descrizione

Interfaccia che rappresenta il contratto attraverso cui implementare la view della pagina di visualizzazione dei capitoli e catturarne gli eventi.

3.13.4.2 Utilizzo

Viene usata per definire il contratto di interazione con l'utente nella pagina di visualizzazione dei capitoli dello sceneggiato.

3.13.4.3 Classi figlie

- Sivodim::Drama::View::ListChapterActivity.

3.13.5 ListChapterActivity (class)

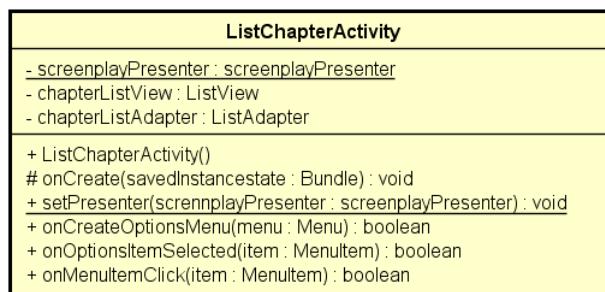


Figura 64: Classe Sivodim::Drama::View::ListChapterActivity

3.13.5.1 Descrizione

Classe che implementa l'interfaccia ListChapterInterface.

3.13.5.2 Utilizzo

Implementa il contratto di interazione con l'utente nella pagina della lista dei capitoli dello sceneggiato.

3.13.5.3 Classi ereditate

- Sivodim::Drama::View::ListChapterInterface.

3.13.5.4 Attributi

3.13.5.4.1 screenplayPresenter

- **Accesso:** private;
- **Tipo:** ScreenplayPresenter;
- **Descrizione:** Riferimento statico al ScreenplayPresenter che la view ListChapterActivity utilizza per interagire con il model, nel rispetto del pattern MVP.

3.13.5.4.2 chapterListView

- **Accesso:** private;
- **Tipo:** ListView;
- **Descrizione:** Componente grafica contenente la lista dei capitoli contenuti all'interno di uno sceneggiato.

3.13.5.4.3 chapterListAdapter

- **Accesso:** private;
- **Tipo:** ListAdapter;
- **Descrizione:** ListAdapter contenente la lista dinamica di tutti i capitoli di tipo Chapter contenuti all'interno di uno sceneggiato.

3.13.5.5 Metodi

3.13.5.5.1 onCreate(Bundle savedInstanceState)

- **Accesso:** protected;
- **Tipo di ritorno:** void;
- **Descrizione:** Inizializza l'activity prelevando le informazioni necessarie dal presenter e predisponendo i listeners previsti.

3.13.6 NewScreenplayInterface (interface)



Figura 65: Classe Sivodim::Drama::View::NewScreenplayInterface

3.13.6.1 Descrizione

Interfaccia che rappresenta il contratto attraverso cui implementare la view della pagina per la creazione di un nuovo sceneggiato e catturarne gli eventi.

3.13.6.2 Utilizzo

Viene usata per definire il contratto di interazione con l'utente nella pagina per la creazione di un nuovo sceneggiato.

3.13.6.3 Classi figlie

- Sivodim::Drama::View::NewScreenplayActivity.

3.13.7 NewScreenplayActivity (class)

NewScreenplayActivity	
- buttonCreateProject : Button	
- editTextProjectName : EditText	
- spinnerImportCharacters : Spinner	
- screenplayPresenter : ScreenplayPresenter	
+ NewScreenplayActivity()	
# onCreate(savedInstanceState : Bundle) : void	
+ onClick(v : View) : void	
+ onOptionItemSelected(item : MenuItem) : boolean	

Figura 66: Classe Sivodim::Drama::View::NewScreenplayActivity

3.13.7.1 Descrizione

Classe che implementa l'interfaccia NewScreenplayInterface.

3.13.7.2 Utilizzo

Implementa il contratto di interazione con l'utente nella pagina per la creazione di un nuovo sceneggiato.

3.13.7.3 Classi ereditate

- Sivodim::Drama::View::NewScreenplayInterface.

3.13.7.4 Attributi

3.13.7.4.1 buttonCreateProject

- **Accesso:** private;
- **Tipo:** Button;
- **Descrizione:** Componente grafica "bottone" necessaria per confermare la creazione di un nuovo sceneggiato.

3.13.7.4.2 editTextProjectName

- **Accesso:** private;
- **Tipo:** EditText;
- **Descrizione:** Componente grafica per l'inserimento di testo che viene prelevato come nome del nuovo capitolo da creare.

3.13.7.4.3 spinnerImportCharacter

- **Accesso:** private;
- **Tipo:** Spinner;
- **Descrizione:** Componente grafica (menu a tendina) contenente la lista di sceneggiati da cui è possibile importare la relativa lista di personaggi.

3.13.7.4.4 screenplayPresenter

- **Accesso:** private;
- **Tipo:** ScreenplayPresenter;
- **Descrizione:** Riferimento statico al ScreenplayPresenter che la view NewScreenplayActivity utilizza per interagire con il model, nel rispetto del pattern MVP.

3.13.7.5 Metodi

3.13.7.5.1 onCreate(Bundle savedInstanceState)

- **Accesso:** protected;
- **Tipo di ritorno:** void;
- **Descrizione:** Inizializza l'activity prelevando le informazioni necessarie dal presenter e predisponendo i listeners previsti.

3.13.8 ListSpeechesInterface (interface)



Figura 67: Classe Sivodim::Drama::View::ListSpeechesInterface

3.13.8.1 Descrizione

Interfaccia che rappresenta il contratto attraverso cui implementare la view della pagina che mostra l'elenco delle battute del capitolo e catturarne gli eventi.

3.13.8.2 Utilizzo

Viene usata per definire il contratto di interazione con l'utente nella pagina che mostra l'elenco delle battute del capitolo.

3.13.8.3 Classi figlie

- Sivodim::Drama::View::ListSpeechesActivity.

3.13.9 NewChapterInterface (interface)



Figura 68: Classe Sivodim::Drama::View::NewChapterInterface

3.13.9.1 Descrizione

Interfaccia che rappresenta il contratto attraverso cui implementare la view della pagina per la creazione di un nuovo capitolo e catturarne gli eventi.

3.13.9.2 Utilizzo

Viene usata per definire il contratto di interazione con l'utente nella pagina per la creazione di un nuovo capitolo.

3.13.9.3 Classi figlie

- Sivodim::Drama::View::NewChapterActivity.

3.13.10 NewChapterActivity (class)

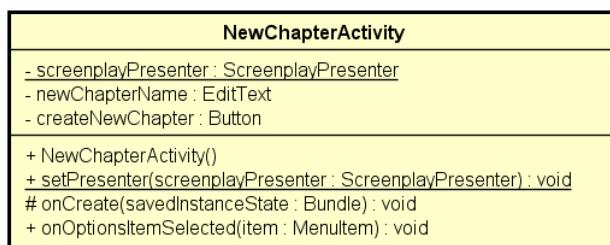


Figura 69: Classe Sivodim::Drama::View::NewChapterActivity

3.13.10.1 Descrizione

Classe che implementa l'interfaccia NewChapterInterface.

3.13.10.2 Utilizzo

Implementa il contratto di interazione con l'utente nella pagina per la creazione di un nuovo capitolo.

3.13.10.3 Classi ereditate

- Sivodim::Drama::View::NewChapterInterface.

3.13.10.4 Attributi

3.13.10.4.1 screenplayPresenter

- **Accesso:** private;
- **Tipo:** ScreenplayPresenter;
- **Descrizione:** Riferimento statico al ScreenplayPresenter che la view NewChapterActivity utilizza per interagire con il model, nel rispetto del pattern MVP.

3.13.10.4.2 newChapterName

- **Accesso:** private;
- **Tipo:** EditText;
- **Descrizione:** Componente grafica necessaria per inserire del testo che viene prelevato come nome del nuovo capitolo.

3.13.10.4.3 createNewChapter

- **Accesso:** private;
- **Tipo:** Button;
- **Descrizione:** Componente grafica "bottone" necessario per confermare la creazione di un nuovo capitolo.

3.13.10.5 Metodi

3.13.10.5.1 onCreate(Bundle savedInstanceState)

- **Accesso:** protected;
- **Tipo di ritorno:** void;
- **Descrizione:** Inizializza l'activity prelevando le informazioni necessarie dal presenter e predisponendo i listeners previsti.

3.13.11 EditChapterInterface (interface)



Figura 70: Classe Sivodim::Drama::View::EditChapterInterface

3.13.11.1 Descrizione

Interfaccia che rappresenta il contratto attraverso cui implementare la view della pagina per la modifica del capitolo di uno sceneggiato e catturarne gli eventi.

3.13.11.2 Utilizzo

Viene usata per definire il contratto di interazione con l'utente nella pagina per la modifica del capitolo di uno sceneggiato.

3.13.11.3 Classi figlie

- Sivodim::Drama::View::EditChapterActivity.

3.13.12 EditChapterActivity (class)

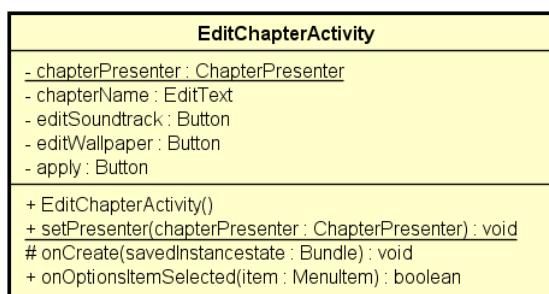


Figura 71: Classe Sivodim::Drama::View::EditChapterActivity

3.13.12.1 Descrizione

Classe che implementa l'interfaccia EditChapterInterface.

3.13.12.2 Utilizzo

Implementa il contratto di interazione con l'utente nella pagina per la modifica di un capitolo di uno sceneggiato.

3.13.12.3 Classi ereditate

- Sivodim::Drama::View::EditChapterInterface.

3.13.12.4 Attributi

3.13.12.4.1 chapterPresenter

- **Accesso:** private;
- **Tipo:** ChapterPresenter;
- **Descrizione:** Riferimento statico al ChapterPresenter che la view EditChapterActivity utilizza per interagire con il model, nel rispetto del pattern MVP.

3.13.12.4.2 chapterName

- **Accesso:** private;
- **Tipo:** EditText;
- **Descrizione:** Componente grafica che visualizza l'attuale nome del capitolo e che può essere modificata per cambiare il nome del capitolo stesso.

3.13.12.4.3 editSoundtrack

- **Accesso:** private;
- **Tipo:** Button;
- **Descrizione:** Componente grafica "bottone" per navigare nella memoria del telefono e selezionare una traccia audio da associare come sottofondo musicale al capitolo.

3.13.12.4.4 editWallpaper

- **Accesso:** private;
- **Tipo:** Button;
- **Descrizione:** Componente grafica "bottone" per navigare nella memoria del dispositivo e selezionare un file immagine da associare come sfondo al capitolo.

3.13.12.4.5 apply

- **Accesso:** private;
- **Tipo:** Button;
- **Descrizione:** Componente grafica "bottone" necessaria per confermare le modifiche apportate al capitolo.

3.13.12.5 Metodi

3.13.12.5.1 onCreate(Bundle savedInstanceState)

- **Accesso:** protected;
- **Tipo di ritorno:** void;
- **Descrizione:** Inizializza l'activity prelevando le informazioni necessarie dal presenter e predisponendo i listeners previsti.

3.13.13 NewSpeechInterface (interface)



Figura 72: Classe Sivodim::Drama::View::NewSpeechInterface

3.13.13.1 Descrizione

Interfaccia che rappresenta il contratto attraverso cui implementare la view della pagina per la creazione di una nuova battuta e catturarne gli eventi.

3.13.13.2 Utilizzo

Viene usata per definire il contratto di interazione con l'utente nella pagina per la creazione di una nuova battuta.

3.13.13.3 Classi figlie

- Sivodim::Drama::View::NewSpeechActivity.

3.13.14 NewSpeechActivity (class)

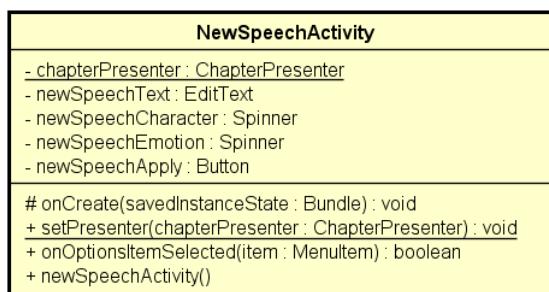


Figura 73: Classe Sivodim::Drama::View::NewSpeechActivity

3.13.14.1 Descrizione

Classe che implementa l'interfaccia NewSpeechInterface.

3.13.14.2 Utilizzo

Implementa il contratto di interazione con l'utente nella pagina per la creazione di una nuova battuta.

3.13.14.3 Classi ereditate

- Sivodim::Drama::View::NewSpeechInterface.

3.13.14.4 Attributi

3.13.14.4.1 chapterPresenter

- **Accesso:** private;
- **Tipo:** ChapterPresenter;
- **Descrizione:** Riferimento statico al ChapterPresenter che la view NewSpeechActivity utilizza per interagire con il model, nel rispetto del pattern MVP.

3.13.14.4.2 newSpeechText

- **Accesso:** private;
- **Tipo:** EditText;
- **Descrizione:** Componente grafica che permette di inserire del testo per creare una nuova battuta.

3.13.14.4.3 newSpeechCharacter

- **Accesso:** private;
- **Tipo:** Spinner;
- **Descrizione:** Componente grafica (menu a tendina) contenente la lista dei personaggi associabili alla nuova battuta. Una battuta può avere un solo personaggio.

3.13.14.4.4 newSpeechEmotion

- **Accesso:** private;
- **Tipo:** Spinner;
- **Descrizione:** Componente grafica (menu a tendina) contenente la lista delle emozioni associabili alla nuova battuta. Una battuta può avere una sola emozione.

3.13.14.4.5 newSpeechApply

- **Accesso:** private;
- **Tipo:** Button;
- **Descrizione:** Componente grafica "bottone" necessaria per confermare la creazione di una nuova battuta.

3.13.14.5 Metodi

3.13.14.5.1 `onCreate(Bundle savedInstanceState)`

- **Accesso:** protected;
- **Tipo di ritorno:** void;
- **Descrizione:** Inizializza l'activity prelevando le informazioni necessarie dal presenter e predisponendo i listeners previsti.

3.13.15 `EditSpeechInterface (interface)`



Figura 74: Classe Sivodim::Drama::View::EditSpeechInterface

3.13.15.1 Descrizione

Interfaccia che rappresenta il contratto attraverso cui implementare la view della pagina per la modifica di una battuta dello sceneggiato e catturarne gli eventi.

3.13.15.2 Utilizzo

Viene usata per definire il contratto di interazione con l'utente nella pagina per la modifica di una battuta dello sceneggiato.

3.13.15.3 Classi figlie

- Sivodim::Drama::View::EditSpeechActivity.

3.13.16 `EditSpeechActivity (class)`

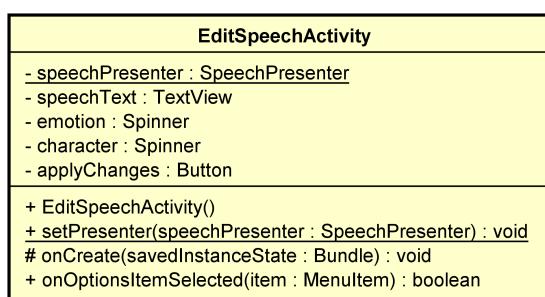


Figura 75: Classe Sivodim::Drama::View::EditSpeechActivity

3.13.16.1 Descrizione

Classe che implementa l'interfaccia EditSpeechInterface.

3.13.16.2 Utilizzo

Implementa il contratto di interazione con l'utente nella pagina per la modifica della battuta di uno sceneggiato.

3.13.16.3 Classi ereditate

- Sivodim::Drama::View::EditSpeechInterface.

3.13.16.4 Attributi

3.13.16.4.1 speechPresenter

- **Accesso:** private;
- **Tipo:** SpeechPresenter;
- **Descrizione:** Riferimento statico al SpeechPresenter che la view EditSpeechActivity utilizza per interagire con il model, nel rispetto del pattern MVP.

3.13.16.4.2 speechText

- **Accesso:** private;
- **Tipo:** TextView;
- **Descrizione:** Componente grafica che visualizza l'attuale testo della battuta e il cui contenuto può essere modificato per cambiare il testo della battuta stessa.

3.13.16.4.3 emotion

- **Accesso:** private;
- **Tipo:** Spinner;
- **Descrizione:** Componente grafica (menu a tendina) contenente la lista di emozioni associabili a una battuta.

3.13.16.4.4 character

- **Accesso:** private;
- **Tipo:** Spinner;
- **Descrizione:** Componente grafica (menu a tendina) contenente la lista di personaggi selezionabili per la battuta. Ogni battuta può avere un solo personaggio.

3.13.16.4.5 applyChanges

- **Accesso:** private;
- **Tipo:** Button;
- **Descrizione:** Componente grafica "bottone" che permette di applicare le modifiche apportate alla battuta.

3.13.16.5 Metodi

3.13.16.5.1 `onCreate(Bundle savedInstanceState)`

- **Accesso:** protected;
- **Tipo di ritorno:** void;
- **Descrizione:** Inizializza l'activity prelevando le informazioni necessarie dal presenter e predisponendo i listeners previsti.

3.13.17 NewCharacterInterface (interface)



Figura 76: Classe Sivodim::Drama::View::NewCharacterInterface

3.13.17.1 Descrizione

Interfaccia che rappresenta il contratto attraverso cui implementare la view della pagina che permette la creazione di un nuovo personaggio e catturarne gli eventi.

3.13.17.2 Utilizzo

Viene usata per definire il contratto di interazione con l'utente nella pagina che permette la creazione di un nuovo personaggio.

3.13.17.3 Classi figlie

- Sivodim::Drama::View::NewCharacterActivity.

3.13.18 NewCharacterActivity (class)

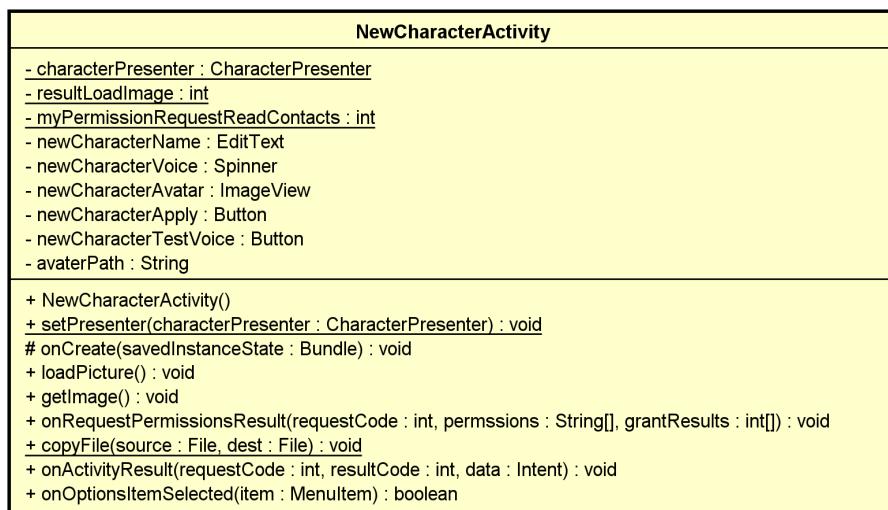


Figura 77: Classe Sivodim::Drama::View::NewCharacterActivity

3.13.18.1 Descrizione

Classe che implementa l’interfaccia NewCharacterInterface.

3.13.18.2 Utilizzo

Implementa il contratto di interazione con l’utente nella pagina per la creazione di un nuovo personaggio.

3.13.18.3 Classi ereditate

- Sivodim::Drama::View::NewCharacterInterface.

3.13.18.4 Attributi

3.13.18.4.1 characterPresenter

- **Accesso:** null;
- **Tipo:** CharacterPresenter;
- **Descrizione:** Riferimento statico al CharacterPresenter che la view NewCharacterActivity utilizza per interagire con il model, nel rispetto del pattern MVP.

3.13.18.4.2 newCharacterName

- **Accesso:** private;
- **Tipo:** EditText;

- **Descrizione:** Componente grafica che permette all'utente di inserire il testo che viene prelevato come nome del nuovo personaggio.

3.13.18.4.3 newCharacterVoice

- **Accesso:** private;
- **Tipo:** Spinner;
- **Descrizione:** Componente grafica (menu a tendina) che permette all'utente di selezionare una voce da associare al nuovo personaggio.

3.13.18.4.4 newCharacterAvatar

- **Accesso:** private;
- **Tipo:** ImageView;
- **Descrizione:** Componente grafica che permette di accedere al caricamento di un avatar per il nuovo personaggio e mostra un'anteprima del caricamento su schermo.

3.13.18.4.5 newCharacterApply

- **Accesso:** private;
- **Tipo:** Button;
- **Descrizione:** Componente grafica "bottone" che permette di confermare i dati inseriti e creare un nuovo personaggio relativamente a uno specifico sceneggiato.

3.13.18.4.6 newCharacterTestVoice

- **Accesso:** private;
- **Tipo:** Button;
- **Descrizione:** Componente grafica "bottone" che permette di avviare un'anteprima della voce scelta per il personaggio durante la creazione.

3.13.18.4.7 avatarPath

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Stringa che contiene il percorso del file immagine associato come avatar_G al personaggio.

3.13.18.4.8 myPermissionsRequestReadMemory

- **Accesso:** private;
- **Tipo:** int;
- **Descrizione:** Intero statico e costante per identificare una risposta a un intent: viene usato quando l'utente risponde al *popup* di richiesta per i permessi necessari per accedere alla memoria.

3.13.18.4.9 resultLoadImage

- **Accesso:** private;
- **Tipo:** int;
- **Descrizione:** Intero statico e costante per identificare una risposta a un intent: viene usato durante la selezione di un'immagine dalla memoria.

3.13.18.5 Metodi

3.13.18.5.1 onCreate(Bundle savedInstanceState)

- **Accesso:** protected;
- **Tipo di ritorno:** void;
- **Descrizione:** Inizializza l'activity prelevando le informazioni necessarie dal presenter e predisponendo i listeners previsti.

3.13.19 ListCharacterInterface (interface)



Figura 78: Classe Sivodim::Drama::View::ListCharacterInterface

3.13.19.1 Descrizione

Interfaccia che rappresenta il contratto attraverso cui implementare la view della pagina che mostra l'elenco dei personaggi dello sceneggiato e catturarne gli eventi.

3.13.19.2 Utilizzo

Viene usata per definire il contratto di interazione con l'utente nella pagina che mostra l'elenco dei personaggi dello sceneggiato.

3.13.19.3 Classi figlie

- Sivodim::Drama::View::ListCharacterActivity.

3.13.20 ListCharacterActivity (class)

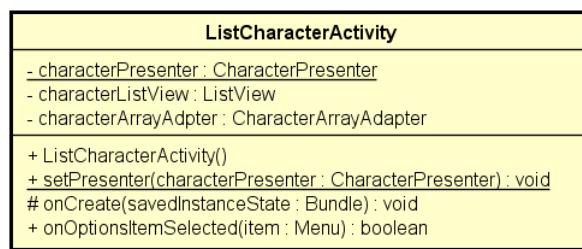


Figura 79: Classe Sivodim::Drama::View::ListCharacterActivity

3.13.20.1 Descrizione

Classe che implementa l'interfaccia ListCharacterInterface.

3.13.20.2 Utilizzo

Implementa il contratto di interazione con l'utente nella pagina che mostra l'elenco dei personaggi di uno sceneggiato.

3.13.20.3 Classi ereditate

- Sivodim::Drama::View::ListCharacterInterface.

3.13.20.4 Attributi

3.13.20.4.1 characterPresenter

- **Accesso:** private;
- **Tipo:** CharacterPresenter;
- **Descrizione:** Riferimento statico al CharacterPresenter che la view ListCharacterActivity utilizza per interagire con il model, nel rispetto del pattern MVP.

3.13.20.4.2 characterListView

- **Accesso:** private;
- **Tipo:** ListView;
- **Descrizione:** Componente grafica che permette di visualizzare la lista di tutti i personaggi relativi a un certo sceneggiato.

3.13.20.4.3 characterArrayAdapter

- **Accesso:** private;
- **Tipo:** CharacterArrayAdapter;
- **Descrizione:** Riferimento ad ArrayAdapter (prelevato dal CharacterPresenter) da associare alla ListView per visualizzare dinamicamente nella GUI i dati relativi a ciascun personaggio.

3.13.20.5 Metodi

3.13.20.5.1 onCreate(Bundle savedInstanceState)

- **Accesso:** protected;
- **Tipo di ritorno:** void;
- **Descrizione:** Inizializza l'activity prelevando le informazioni necessarie dal presenter e predisponendo i listeners previsti.

3.13.21 EditCharacterInterface (interface)



Figura 80: Classe Sivodim::Drama::View::EditCharacterInterface

3.13.21.1 Descrizione

Interfaccia che rappresenta il contratto attraverso cui implementare la view della pagina per la modifica di un personaggio dello sceneggiato e catturarne gli eventi.

3.13.21.2 Utilizzo

Viene usata per definire il contratto di interazione con l'utente nella pagina per la modifica di un personaggio dello sceneggiato.

3.13.21.3 Classi figlie

- Sivodim::Drama::View::EditCharacterActivity.

3.13.22 EditCharacterActivity (class)

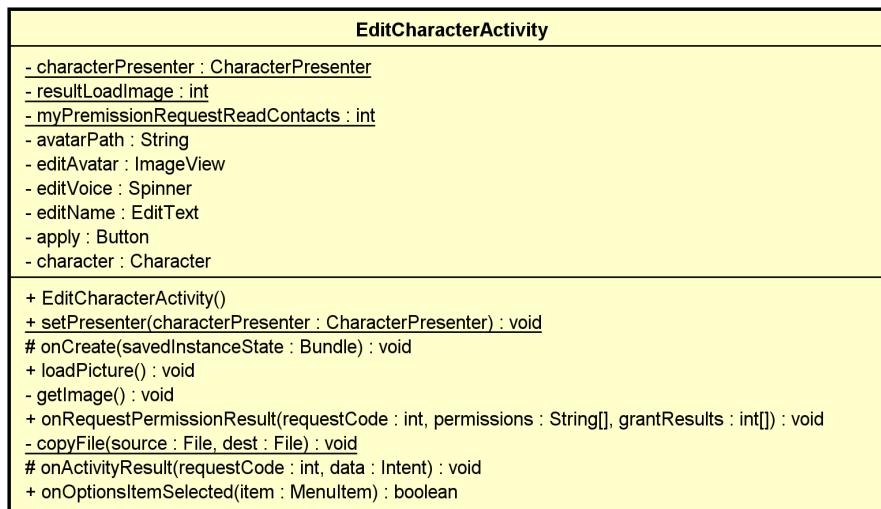


Figura 81: Classe Sivodim::Drama::View::EditCharacterActivity

3.13.22.1 Descrizione

Classe che implementa l'interfaccia EditCharacterInterface.

3.13.22.2 Utilizzo

Implementa il contratto dell'interfaccia EditCharacterInterface, necessaria per permettere all'utente di interagire con la schermata di modifica di un dato personaggio.

3.13.22.3 Classi ereditate

- Sivodim::Drama::View::EditCharacterInterface.

3.13.22.4 Attributi

3.13.22.4.1 characterPresenter

- **Accesso:** private;
- **Tipo:** CharacterPresenter;
- **Descrizione:** Riferimento statico al CharacterPresenter che la view EditCharacterActivity utilizza per interagire con il model, nel rispetto del pattern MVP.

3.13.22.4.2 avatarPath

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Stringa che contiene il percorso dell'avatar_G relativo al personaggio.

3.13.22.4.3 editAvatar

- **Accesso:** private;
- **Tipo:** ImageView;
- **Descrizione:** Componente grafica che mostra un'anteprima dell'avatar e permette di accedere alla modifica dello stesso.

3.13.22.4.4 editVoice

- **Accesso:** private;
- **Tipo:** Spinner;
- **Descrizione:** Componente grafica che mostra un menu a tendina contenente la lista di voci selezionabili per il personaggio.

3.13.22.4.5 editName

- **Accesso:** private;
- **Tipo:** EditText;
- **Descrizione:** Componente grafica contenente il nome attuale del personaggio e il cui testo può essere modificato per cambiare il nome al personaggio stesso.

3.13.22.4.6 apply

- **Accesso:** private;
- **Tipo:** Button;
- **Descrizione:** Componente grafica "bottone" per applicare le modifiche apportate ai dati del personaggio mostrati nell'Activity.

3.13.22.4.7 character

- **Accesso:** private;
- **Tipo:** Character;
- **Descrizione:** Riferimento al personaggio prelevato dal CharacterPresenter i cui dati vengono manipolati all'interno dell'Activity.

3.13.22.4.8 resultLoadImage

- **Accesso:** private;
- **Tipo:** int;
- **Descrizione:** Intero statico e costante per identificare una risposta a un intent: viene usato durante la selezione di un'immagine dalla memoria.

3.13.22.4.9 myPermissionsRequestReadMemory

- **Accesso:** private;
- **Tipo:** int;
- **Descrizione:** Intero statico e costante per identificare una risposta a un intent: viene usato quando l'utente risponde al *popup* di richiesta per i permessi necessari per accedere alla memoria.

3.13.22.5 Metodi

3.13.22.5.1 onCreate(Bundle savedInstanceState)

- **Accesso:** protected;
- **Tipo di ritorno:** void;
- **Descrizione:** Inizializza l'activity prelevando le informazioni necessarie dal presenter e predisponendo i listeners previsti.

3.13.23 ListSpeechesActivity (class)

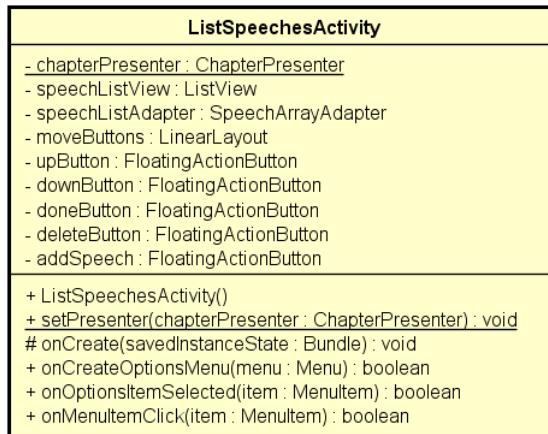


Figura 82: Classe Sivodim::Drama::View::ListSpeechesActivity

3.13.23.1 Descrizione

Classe che implementa l'interfaccia ListSpeechesInterface.

3.13.23.2 Utilizzo

Implementa il contratto di interazione con l'utente nella pagina per la visualizzazione delle battute di uno sceneggiato.

3.13.23.3 Classi ereditate

- Sivodim::Drama::View::ListSpeechesInterface.

3.13.23.4 Attributi

3.13.23.4.1 chapterPresenter

- **Accesso:** private;
- **Tipo:** ChapterPresenter;
- **Descrizione:** Riferimento statico al ChapterPresenter che la view ListSpeechesActivity utilizza per interagire con il model, nel rispetto del pattern MVP.

3.13.23.4.2 speechListView

- **Accesso:** private;
- **Tipo:** ListView;
- **Descrizione:** Componente grafica che permette di visualizzare sotto forma di lista le battute contenute all'interno di un capitolo.

3.13.23.4.3 speechListAdapter

- **Accesso:** private;
- **Tipo:** SpeechArrayAdapter;
- **Descrizione:** ArrayAdapter contenente la lista dinamica delle battute e a cui viene associata la componente grafica speechListView.

3.13.23.4.4 moveButtons

- **Accesso:** private;
- **Tipo:** LinearLayout;
- **Descrizione:** Componente grafica che racchiude i bottoni necessari al riordinamento e all'eliminazione delle battute. Il layout e i bottoni contenuti in esso compaiono dopo la pressione prolungata di una componente grafica battuta.

3.13.23.4.5 upButton

- **Accesso:** private;
- **Tipo:** FloatingActionButton;
- **Descrizione:** Componente grafica "bottone" utile allo spostamento verso l'alto della battuta selezionata.

3.13.23.4.6 downButton

- **Accesso:** private;
- **Tipo:** FloatingActionButton;
- **Descrizione:** Componente grafica "bottone" utile allo spostamento verso il basso della battuta selezionata.

3.13.23.4.7 doneButton

- **Accesso:** private;
- **Tipo:** FloatingActionButton;
- **Descrizione:** Componente grafica "bottone" necessaria per la conferma di uno spostamento eseguito sulla battuta.

3.13.23.4.8 deleteButton

- **Accesso:** private;
- **Tipo:** FloatingActionButton;
- **Descrizione:** Componente grafica "bottone" necessaria per l'eliminazione della battuta selezionata.

3.13.23.4.9 addSpeech

- **Accesso:** private;
- **Tipo:** FloatingActionButton;
- **Descrizione:** Componente grafica "bottone" posta in basso a destra dello schermo necessaria per l'aggiunta di una nuova battuta.

3.13.23.5 Metodi

3.13.23.5.1 ListSpeechesActivity()

- **Accesso:** public;
- **Descrizione:** Costruttore di *default*.

3.13.23.5.2 setPresenter(ChapterPresenter chapterPresenter)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - chapterPresenter: presenter che deve gestire l'activity.
- **Descrizione:** Assegna il presenter staticamente alla Activity.

3.14 Sivodim::Drama::Model::Utilities

3.14.1 Informazioni sul package

Di seguito viene illustrata, tramite un diagramma delle classi, la visione di insieme del package Drama::Model::Utilities contenente tutte le classi necessarie per la gestione di file audio, file immagine e per il salvataggio in xml degli sceneggiati.

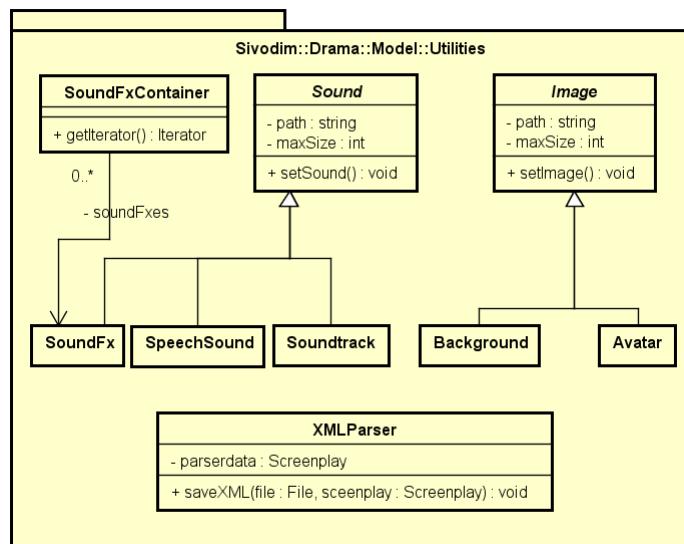


Figura 83: Componente Sivodim::Drama::Model::Utilities

3.14.2 Sound (abstract)

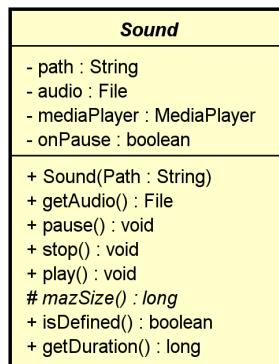


Figura 84: Classe Sivodim::Drama::Model::Utilities::Sound

3.14.2.1 Descrizione

Classe astratta che rappresenta un file audio generico. Viene concretizzata nelle classi che rappresentano un effetto sonoro, l'audio di una battuta e una colonna sonora.

3.14.2.2 Utilizzo

Viene usata per rappresentare un file audio di tipo generico.

3.14.2.3 Classi figlie

- Sivodim::Drama::Model::Utilities::SoundFx;
- Sivodim::Drama::Model::Utilities::SpeechSound;
- Sivodim::Drama::Model::Utilities::Soundtrack.

3.14.2.4 Attributi

3.14.2.4.1 path

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Stringa che rappresenta il percorso (in memoria SD) del file audio.

3.14.2.4.2 audio

- **Accesso:** private;
- **Tipo:** File;
- **Descrizione:** Oggetto contenente i dati del File audio presente nella memoria SD del dispositivo.

3.14.2.4.3 mediaplayer

- **Accesso:** private;
- **Tipo:** Medioplayer;
- **Descrizione:** Oggetto che rappresenta il *file* audio in modo da poterne gestire le azioni di avvio, pausa e stop.

3.14.2.4.4 onPause

- **Accesso:** private;
- **Tipo:** boolean;
- **Descrizione:** flag per mediaPlayer.

3.14.2.5 Metodi

3.14.2.5.1 Sound(String path)

- **Accesso:** public;
- **Argomenti:**
 - path: il percorso al file audio.
- **Descrizione:** Costruttore che inizializza this.path e crea un file a partire dal path passato per parametro, se questo non supera la dimensione massima prevista dalla funzione astratta maxSize().

3.14.2.5.2 getAudio()

- **Accesso:** public;
- **Tipo di ritorno:** File;
- **Descrizione:** Restituisce il file memorizzato in this.audio;
- **Note:** Può restituire null.

3.14.2.5.3 pause()

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Descrizione:** Mette in pausa la riproduzione del file audio memorizzato, se si stava riproducendo.

3.14.2.5.4 stop()

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Descrizione:** Ferma la riproduzione di un file audio se si stava riproducendo.

3.14.2.5.5 play()

- **Accesso:** null;
- **Tipo di ritorno:** void;
- **Descrizione:** Inizia la riproduzione del file audio memorizzato in this.audio, eventualmente riprendendo dall'ultima pausa.

3.14.2.5.6 maxSize()

- **Accesso:** protected;
- **Tipo di ritorno:** long;
- **Descrizione:** Metodo astratto che deve essere definito nelle sottoclassi per ritornare la dimensione massima che il file audio potrà avere;
- **Note:** Determina la costruzione dell'oggetto.

3.14.2.5.7 isDefined()

- **Accesso:** public;
- **Tipo di ritorno:** boolean;
- **Descrizione:** Restituisce true se e solo se this.audio è stato inizializzato ed il file che memorizza esiste.

3.14.2.5.8 getDuration()

- **Accesso:** public;
- **Tipo di ritorno:** long;
- **Descrizione:** Restituisce la durata in millisecondi di un audio.

3.14.3 SoundFx (class)

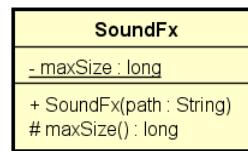


Figura 85: Classe Sivodim::Drama::Model::Utilities::SoundFx

3.14.3.1 Descrizione

Classe che implementa la rappresentazione di un oggetto di tipo SoundFx, ossia un effetto sonoro.

3.14.3.2 Utilizzo

Viene usata per concretizzare la classe Sound in un oggetto di tipo SoundFx.

3.14.3.3 Classi ereditate

- Sivodim::Drama::Model::Utilities::Sound.

3.14.3.4 Attributi

3.14.3.4.1 maxSize

- **Accesso:** private;
- **Tipo:** long;
- **Descrizione:** Intero statico e costante che definisce la grandezza massima di un effetto sonoro: 15MB.

3.14.3.5 Metodi

3.14.3.5.1 SoundFx(String path)

- **Accesso:** public;
- **Argomenti:**
 - path: il percorso al file audio.
- **Descrizione:** Costruttore che prende per parametro il path del file audio (effetto sonoro) che SoundFx rappresenta.

3.14.3.5.2 maxSize()

- **Accesso:** protected;
- **Tipo di ritorno:** long;
- **Descrizione:** Definizione del metodo astratto derivato da Sound, restituisce la dimensione massima che un file audio SoundFx può avere.

3.14.4 SpeechSound (class)



Figura 86: Classe Sivodim::Drama::Model::Utilities::SpeechSound

3.14.4.1 Descrizione

Classe che implementa un oggetto di tipo SpeechSound, ossia l'audio sintetizzato dal testo di una battuta.

3.14.4.2 Utilizzo

Viene usata per concretizzare la classe Sound in un oggetto nel tipo SpeechSound.

3.14.4.3 Classi ereditate

- Sivodim::Drama::Model::Utilities::Sound.

3.14.4.4 Attributi

3.14.4.4.1 maxSize

- **Accesso:** private;
- **Tipo:** long;
- **Descrizione:** Intero statico e costante che definisce la grandezza massima di un file audio di sottofondo: 15MB.

3.14.4.5 Metodi

3.14.4.5.1 SpeechSound(String path)

- **Accesso:** public;
- **Argomenti:**
 - path: il percorso al file audio.
- **Descrizione:** Costruttore che prende per parametro il path del file audio generato a partire da un oggetto di tipo Speech (la sintesi di una battuta).

3.14.4.5.2 maxSize()

- **Accesso:** protected;
- **Tipo di ritorno:** long;
- **Descrizione:** Definizione del metodo astratto derivato da Sound, restituisce la dimensione massima che un file audio SpeechSound può avere.

3.14.5 Soundtrack (class)



Figura 87: Classe Sivodim::Drama::Model::Utilities::Soundtrack

3.14.5.1 Descrizione

Classe che implementa un oggetto di tipo Soundtrack, ossia la colonna sonora relativa ad un capitolo.

3.14.5.2 Utilizzo

Viene usata per concretizzare la classe Sound in un oggetto del tipo Soundtrack.

3.14.5.3 Classi ereditate

- Sivodim::Drama::Model::Utilities::Sound.

3.14.5.4 Attributi

3.14.5.4.1 maxSize

- **Accesso:** private;
- **Tipo:** long;
- **Descrizione:** Intero statico e costante che definisce la grandezza massima di un file audio di sottofondo: 15MB.

3.14.5.5 Metodi

3.14.5.5.1 Soundtrack(String path)

- **Accesso:** public;
- **Argomenti:**
 - path: il percorso al file audio.
- **Descrizione:** Costruttore che prende per parametro il path del file audio che farà da sottofondo sonoro.

3.14.5.5.2 maxSize()

- **Accesso:** protected;
- **Tipo di ritorno:** long;
- **Descrizione:** Definizione del metodo astratto derivato da Sound, restituisce la dimensione massima che un file audio Soundtrack può avere.

3.14.6 Image (abstract)

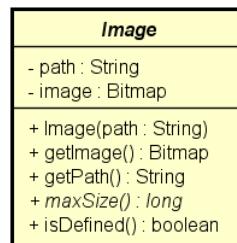


Figura 88: Classe Sivodim::Drama::Model::Utilities::Image

3.14.6.1 Descrizione

Classe astratta che rappresenta un file di tipo immagine. Viene concretizzata nelle classi che rappresentano un'immagine di sfondo e un *avatar_G* da assegnare a un personaggio.

3.14.6.2 Utilizzo

Viene usata per rappresentare un file di tipo immagine attraverso un oggetto che contiene il percorso del file nella memoria del dispositivo e la sua grandezza massima.

3.14.6.3 Classi figlie

- Sivodim::Drama::Model::Utilities::Background;
- Sivodim::Drama::Model::Utilities::Avatar.

3.14.6.4 Attributi

3.14.6.4.1 path

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Stringa che rappresenta il percorso (nella memoria SD) del file immagine.

3.14.6.4.2 image

- **Accesso:** private;
- **Tipo:** Bitmap;
- **Descrizione:** Oggetto che rappresenta il file immagine durante l'esecuzione dell'applicazione.

3.14.6.5 Metodi

3.14.6.5.1 Image(String path)

- **Accesso:** public;
- **Argomenti:**
 - path: il percorso al file immagine.
- **Descrizione:** Costruttore della classe astratta Image. Prende per parametro il path del file di immagine, lo memorizza, e carica una Bitmap in base al file;
- **Note:** Se la dimensione massima del file, specificata dal metodo astratto maxSize, viene superata la Bitmap non viene creata.

3.14.6.5.2 maxSize()

- **Accesso:** protected;
- **Tipo di ritorno:** long;
- **Descrizione:** Metodo astratto che restituisce la dimensione massima che un File rappresentato dall'oggetto Image può avere.

3.14.7 Background (class)

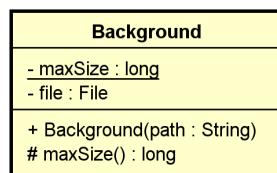


Figura 89: Classe Sivodim::Drama::Model::Utilities::Background

3.14.7.1 Descrizione

Classe che implementa la rappresentazione di un oggetto di tipo Background, ossia un'immagine di sfondo relativa a un capitolo.

3.14.7.2 Utilizzo

Viene usata per concretizzare la classe Image in un oggetto del tipo Background.

3.14.7.3 Classi ereditate

- Sivodim::Drama::Model::Utilities::Image.

3.14.7.4 Attributi

3.14.7.4.1 maxSize

- **Accesso:** private;
- **Tipo:** long;
- **Descrizione:** Intero statico e costante che definisce la grandezza massima di un'immagine di sfondo: 1MB.

3.14.7.5 Metodi

3.14.7.5.1 Backgroud(String path)

- **Accesso:** public;
- **Argomenti:**
 - path: il percorso al file immagine.
- **Descrizione:** Costruttore che prende per parametro il path del file che farà da immagine di sfondo.

3.14.7.5.2 maxSize()

- **Accesso:** protected;
- **Tipo di ritorno:** long;
- **Descrizione:** Definizione del metodo astratto derivato da Image, restituisce la dimensione massima che un file di immagine Background può avere.

3.14.8 Avatar (class)

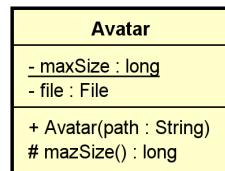


Figura 90: Classe Sivodim::Drama::Model::Utilities::Avatar

3.14.8.1 Descrizione

Classe che implementa la rappresentazione di un oggetto di tipo Avatar.

3.14.8.2 Utilizzo

Viene usata per concretizzare la classe Image in un oggetto del tipo Avatar.

3.14.8.3 Classi ereditate

- Sivodim::Drama::Model::Utilities::Image.

3.14.8.4 Attributi

3.14.8.4.1 maxSize

- **Accesso:** private;
- **Tipo:** long;
- **Descrizione:** Intero statico e costante che definisce la grandezza massima dell'avatar_G associato a un personaggio: 512KB.

3.14.8.5 Metodi

3.14.8.5.1 Avatar(String path)

- **Accesso:** public;
- **Argomenti:**
 - path: il percorso al file immagine.
- **Descrizione:** Costruttore della classe astratta Avatar. Prende per parametro il path del file di immagine che farà da *avatar_G* ad un personaggio.

3.14.8.5.2 maxSize()

- **Accesso:** protected;
- **Tipo di ritorno:** long;
- **Descrizione:** Definizione del metodo astratto derivato da Image, restituisce la dimensione massima che un file di immagine Avatar può avere.

3.14.9 XMLParser (class)

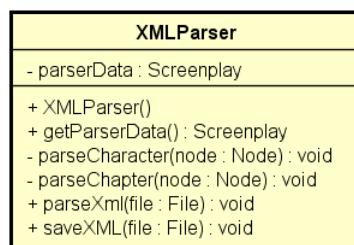


Figura 91: Classe Sivodim::Drama::Model::Utilities::XMLParser

3.14.9.1 Descrizione

Classe di utilità che si occupa di creare file XMLG e farne parsingG.

3.14.9.2 Utilizzo

Viene utilizzata per salvare in memoria oggetti Screenplay sotto forma di files XML e riconvertirli poi in oggetti nella fase di load.

3.14.9.3 Attributi

3.14.9.3.1 parsedData

- **Accesso:** private;
- **Tipo:** Screenplay;
- **Descrizione:** Memorizza lo Screenplay caricato dalla memoria.

3.14.9.4 Metodi

3.14.9.4.1 XMLParser()

- **Accesso:** public;
- **Descrizione:** Costruttore di default.

3.14.9.4.2 getParsedData()

- **Accesso:** public;
- **Tipo di ritorno:** Screenplay;
- **Descrizione:** Restituisce lo Screenplay creato dal metodo parseXML e presente in this.parsedData.

3.14.9.4.3 parseCharacter(Node node)

- **Accesso:** private;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - node: il nodo che contiene il personaggio.
- **Descrizione:** Aggiunge a this.parsedData il personaggio ottenuto dalla conversione del nodo XML passato per parametro.

3.14.9.4.4 parseChapter(Node node)

- **Accesso:** private;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - node: il nodo che contiene il Chapter
- **Descrizione:** Aggiunge a this.parsedData il capitolo ottenuto dalla conversione del nodo XML passato per parametro.

3.14.9.4.5 parseXml(File file)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - file: il file XML da convertire in Screenplay
- **Descrizione:** Salva in this.parsedData lo Screenplay ottenuto dalla conversione del file XML. Per effettuare questa conversione si avvale di parseCharacter e parseChapter. Il parsing viene eseguito attraverso DOM_G.

3.14.9.4.6 saveXML(File file, Screenplay screenplay)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - file: il file in cui salvare lo Screenplay;
 - screenplay: lo Screenplay da salvare.
- **Descrizione:** Converte in XML lo Screenplay passato e lo salva nel file indicato. Il parsing viene eseguito attraverso DOM_G.

3.15 Sivodim::Drama::Model::Chapter

3.15.1 Informazioni sul package

Di seguito viene illustrata, tramite un diagramma delle classi, la visione di insieme del package Drama::Model::Chapter contenente tutte le classi necessarie per la gestione di capitoli e battute.

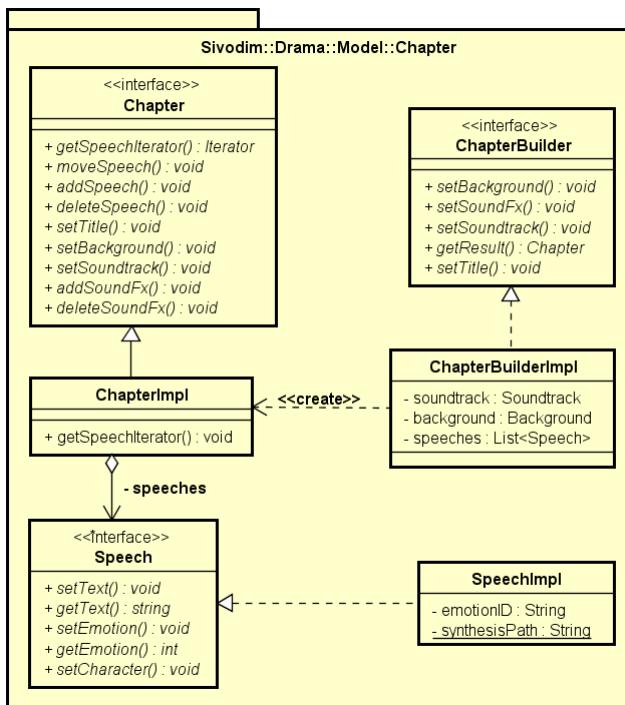


Figura 92: Componente Sivodim::Drama::Model::Chapter

3.15.2 Chapter (interface)

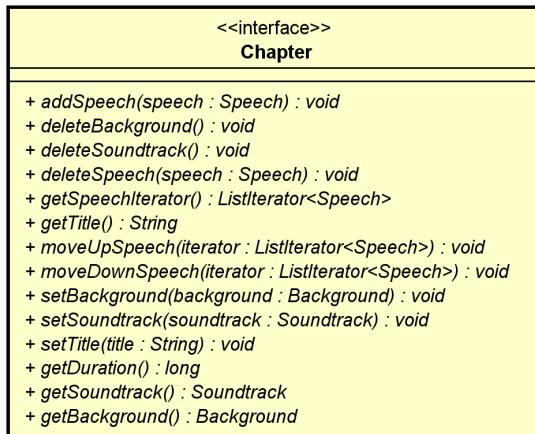


Figura 93: Classe Sivodim::Drama::Model::Chapter::Chapter

3.15.2.1 Descrizione

Interfaccia che definisce il contratto di un oggetto di tipo capitolo. La sua implementazione ChapterImpl ha un riferimento all'interfaccia Speech che definisce il contratto di un oggetto di tipo battuta.

3.15.2.2 Utilizzo

Viene usata per rappresentare un oggetto di tipo capitolo.

3.15.2.3 Classi figlie

- Sivodim::Drama::Model::Chapter::ChapterImpl.

3.15.2.4 Metodi

3.15.2.4.1 getSpeechIterator()

- **Accesso:** public;
- **Tipo di ritorno:** Iterator<Speech>;
- **Descrizione:** Restituisce un iteratore che consente di scorrere fra le battute (oggetti Speech) contenute nel capitolo.

3.15.2.4.2 getTitle()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Restituisce il titolo del capitolo.

3.15.2.4.3 setSoundtrack(Soundtrack soundtrack)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - soundtrack: la Soundtrack da assegnare al capitolo.
- **Descrizione:** Imposta come sottofondo sonoro del capitolo l'oggetto Soundtrack passato per parametro.

3.15.2.4.4 setTitle(String title)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - title: il titolo da assegnare al capitolo.
- **Descrizione:** Imposta il titolo del capitolo con la String passata per parametro.

3.15.2.4.5 setBackground(Background background)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - background: il Background da assegnare al capitolo come sfondo.
- **Descrizione:** Imposta come immagine di sfondo del capitolo l'oggetto di tipo Background ricevuto per parametro.

3.15.2.4.6 addSpeech(Speech speech)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - speech: la battuta da assegnare al capitolo.
- **Descrizione:** Metodo che aggiunge al capitolo la battuta ricevuta per parametro.

3.15.2.4.7 deleteSpeech(Speech speech)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - speech: la battuta da togliere dal capitolo.
- **Descrizione:** Elimina dal capitolo la battuta il cui riferimento è passato per parametro.

3.15.2.4.8 deleteBackground()

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Descrizione:** Elimina dal capitolo lo sfondo.

3.15.2.4.9 deleteSoundtrack()

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Descrizione:** Elimina dal capitolo il riferimento a Soundtrack che costituisce il sotto-fondo sonoro.

3.15.2.4.10 moveUpSpeech(ListIterator<Speech> iterator)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - iterator: l'iteratore che si riferisce alla Speech da portare indietro nell'ordine del capitolo.
- **Descrizione:** Sposta prima una battuta nell'ordine del capitolo. Prende per riferimento l'iteratore che punta alla battuta da spostare;
- **Note:** Se la battuta è all'inizio del capitolo non viene spostata.

3.15.2.4.11 moveDownSpeech(ListIterator<Speech> iterator)

- **Accesso:** null;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - iterator: l'iteratore che punta alla Speech da portare avanti nell'ordine del capitolo.
- **Descrizione:** Sposta in avanti una battuta nell'ordine del capitolo. Prende per riferimento l'iteratore che punta alla battuta da spostare;
- **Note:** Se la battuta è alla fine del capitolo non viene spostata.

3.15.2.4.12 getDuration()

- **Accesso:** public;
- **Tipo di ritorno:** long;
- **Descrizione:** Metodo che ritorna la durata in millisecondi del capitolo sommando la durata di ogni singola battuta contenuta nella lista List<Speech>.

3.15.2.4.13 getSoundtrack()

- **Accesso:** public;
- **Tipo di ritorno:** Soundtrack;
- **Descrizione:** Metodo che ritorna la soundtrack (audio) di un capitolo.

3.15.2.4.14 getBackground()

- **Accesso:** public;
- **Tipo di ritorno:** Background;
- **Descrizione:** Metodo che ritorna la durata il background (immagine) di un capitolo.

3.15.3 ChapterImpl (class)

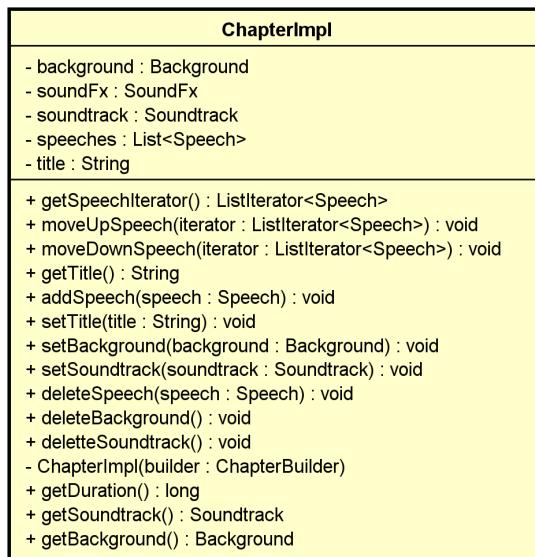


Figura 94: Classe Sivodim::Drama::Model::Chapter::ChapterImpl

3.15.3.1 Descrizione

Classe che implementa la rappresentazione di un oggetto di tipo capitolo.

3.15.3.2 Utilizzo

Viene usata per concretizzare la classe Chapter in un oggetto del tipo ChapterImpl.

3.15.3.3 Classi ereditate

- Sivodim::Drama::Model::Chapter::Chapter.

3.15.3.4 Attributi

3.15.3.4.1 title

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Stringa di testo che rappresenta il titolo del capitolo.

3.15.3.4.2 background

- **Accesso:** private;
- **Tipo:** Background;
- **Descrizione:** Oggetto che rappresenta un file immagine di tipo Background, ossia lo sfondo associato al capitolo.

3.15.3.4.3 soundtrack

- **Accesso:** private;
- **Tipo:** Soundtrack;
- **Descrizione:** Oggetto che rappresenta un file audio di tipo Soundtrack, ossia il sottofondo musicale associato al capitolo.

3.15.3.4.4 soundFx

- **Accesso:** private;
- **Tipo:** SoundFx;
- **Descrizione:** Oggetto che rappresenta un file audio di tipo SoundFx, ossia l'effetto sonoro associato al capitolo.

3.15.3.4.5 speeches

- **Accesso:** private;
- **Tipo:** List<Speech>;
- **Descrizione:** Lista contenente tutte le battute di tipo Speech relative al capitolo.

3.15.3.5 Metodi

3.15.3.5.1 getTitle()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia Chapter restituendo this.title.

3.15.3.5.2 ChapterImpl(ChapterBuilder builder)

- **Accesso:** private;
- **Argomenti:**
 - builder: un builder di Chapter.
- **Descrizione:** Costruttore privato di ChapterImpl che prende per parametro un ChapterBuilder (applicazione del *design pattern* Builder).

3.15.3.5.3 getSpeechIterator()

- **Accesso:** public;
- **Tipo di ritorno:** ListIterator<Speech>;
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia Chapter restituendo un iteratore di Speech. Per farlo utilizza il metodo listIterator del contenitore speeches di tipo List che ha per attributo.

3.15.3.5.4 moveUpSpeech(ListIterator<Speech> iterator)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - iterator: l'iteratore che punta alla battuta da portare indietro nell'ordine del capitolo.
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia Chapter spostando indietro di una posizione nella lista speeches la Speech puntata dall'iteratore passato per parametro.

3.15.3.5.5 moveDownSpeech(ListIterator<Speech> iterator)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - iterator: l'iteratore che punta alla Speech da portare avanti nell'ordine del capitolo.
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia Chapter spostando in avanti di una posizione nella lista speeches la Speech puntata dall'iteratore passato per parametro.

3.15.3.5.6 addSpeech(Speech speech)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - speech: la Speech da aggiungere al capitolo.
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia Chapter aggiungendo in coda a this.speeches la Speech passata per parametro.

3.15.3.5.7 setTitle(String title)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - title: il titolo da dare al capitolo.
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia Chapter memorizzando in this.title la String passata per parametro.

3.15.3.5.8 setBackground(Background background)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - background: lo sfondo da dare al capitolo.
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia Chapter memorizzando in this.background il Background passato per parametro.

3.15.3.5.9 setSoundtrack(Soundtrack soundtrack)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - soundtrack: il sottofondo sonoro da dare al capitolo.
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia Chapter memorizzando in this.soundtrack la Soundtrack passata per parametro.

3.15.3.5.10 deleteSpeech(Speech speech)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - speech: la battuta da togliere dal capitolo.
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia Chapter togliendo da this.speeches il riferimento allo Speech passato per parametro.

3.15.3.5.11 deleteBackground()

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia Chapter settando a null il riferimento this.background.

3.15.3.5.12 deleteSoundtrack()

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia Chapter settando a null il riferimento this.soundtrack.

3.15.3.5.13 getDuration()

- **Accesso:** public;
- **Tipo di ritorno:** long;
- **Descrizione:** Metodo che ritorna la durata in millisecondi del capitolo sommando la durata di ogni singola battuta contenuta nella lista List<Speech>.

3.15.3.5.14 getSoundtrack()

- **Accesso:** public;
- **Tipo di ritorno:** Soundtrack;
- **Descrizione:** Metodo che ritorna la soundtrack (audio) di un capitolo.

3.15.3.5.15 getBackground()

- **Accesso:** public;
- **Tipo di ritorno:** Background;
- **Descrizione:** Metodo che ritorna la durata il background (immagine) di un capitolo.

3.15.4 ChapterBuilder (class)

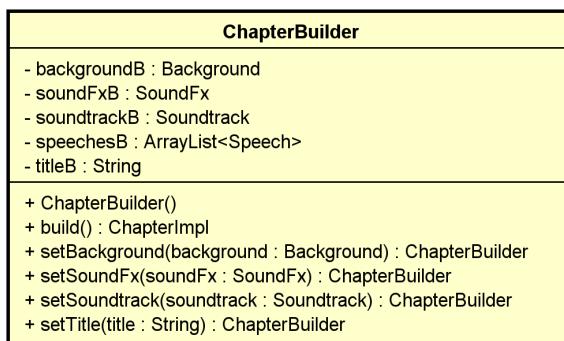


Figura 95: Classe Sivodim::Drama::Model::Chapter::ChapterBuilder

3.15.4.1 Descrizione

Classe interna a ChapterImpl che implementa un oggetto di tipo ChapterBuilder necessario per la creazione di un oggetto ChapterImpl.

3.15.4.2 Utilizzo

Viene usata per implementare il *Builder* necessario per la creazione di un oggetto di tipo capitolo.

3.15.4.3 Classi ereditate

- Sivodim::Drama::Model::Chapter::ChapterBuilder.

3.15.4.4 Attributi

3.15.4.4.1 titleB

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Memorizza il titolo con cui il Chapter verrà creato.

3.15.4.4.2 backgroundB

- **Accesso:** private;
- **Tipo:** Background;
- **Descrizione:** Memorizza lo sfondo con cui il Chapter verrà creato.

3.15.4.4.3 soundtrackB

- **Accesso:** private;
- **Tipo:** Soundtrack;
- **Descrizione:** Memorizza il sottofondo sonoro con cui il Chapter verrà creato.

3.15.4.4.4 speechesB

- **Accesso:** private;
- **Tipo:** ArrayList<Speech>;
- **Descrizione:** Memorizza le battute con cui il Chapter verrà creato.

3.15.4.4.5 soundFxB

- **Accesso:** private;
- **Tipo:** SoundFx;
- **Descrizione:** Memorizza i suoni con cui il Chapter verrà creato.

3.15.4.5 Metodi

3.15.4.5.1 ChapterBuilder()

- **Accesso:** public;
- **Descrizione:** Costruttore di default.

3.15.4.5.2 build()

- **Accesso:** public;
- **Tipo di ritorno:** ChapterImpl;
- **Descrizione:** Costruisce un oggetto di tipo ChapterImpl.

3.15.4.5.3 setTitle(String title)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - title: il titolo da dare al capitolo.
- **Descrizione:** Implementa il metodo dichiarato in ChapterBuilder memorizzando in this.title la String passata per parametro.

3.15.4.5.4 setBackground(Background background)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - background: lo sfondo da dare al capitolo.
- **Descrizione:** Implementa il metodo dichiarato in ChapterBuilder memorizzando in this.background il Background passato per parametro.

3.15.4.5.5 setSoundtrack(Soundtrack soundtrack)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - soundtrack: il sottofondo sonoro da dare al capitolo.
- **Descrizione:** Implementa il metodo dichiarato in ChapterBuilder memorizzando in this.soundtrack la Soundtrack passata per parametro.

3.15.4.5.6 setSoundFx(SoundFx soundFx)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - soundFx: l'effetto sonoro da assegnare al capitolo.
- **Descrizione:** Implementa il metodo dichiarato in ChapterBuilder memorizzando in this.soundFx il SoundFx passato per parametro.

3.15.5 Speech (interface)

<<interface>>	
Speech	
+ getCharacter() : Character	
+ getEmotion() : String	
+ getText() : String	
+ setCharacter(character : Character) : void	
+ setEmotion(emotionID : String) : void	
+ setSoundFx(soundFx : SoundFx) : void	
+ setText(text : String) : void	
+ getAudioPath() : String	
+ getAudioStatus() : boolean	
+ getDuration() : long	
+ setAudioPath(audioPath : String) : void	
+ setAudioStatus(audioStatus : boolean) : void	
+ synthesizeAudio(context : Context, path : String) : void	

Figura 96: Classe Sivodim::Drama::Model::Chapter::Speech

3.15.5.1 Descrizione

Interfaccia che rappresenta il contratto attraverso cui implementare un oggetto di tipo battuta.

3.15.5.2 Utilizzo

Viene usata per definire il contratto di implementazione di un oggetto di tipo battuta.

3.15.5.3 Classi figlie

- Sivodim::Drama::Model::Chapter::SpeechImpl.

3.15.5.4 Metodi

3.15.5.4.1 setText(String text)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - text: il testo da assegnare alla battuta.
- **Descrizione:** Memorizza il testo della battuta che viene passato per parametro String.

3.15.5.4.2 setEmotion(String emotionID)

- **Accesso:** public;
- **Tipo di ritorno:** void;

- **Argomenti:**

- emotionID: il nome che identifica l'emozione da assegnare alla battuta.

- **Descrizione:** Memorizza l'emozione della battuta che viene passata per parametro String.

3.15.5.4.3 setCharacter(Character character)

- **Accesso:** public;

- **Tipo di ritorno:** void;

- **Argomenti:**

- character: il personaggio da associare alla battuta.

- **Descrizione:** Memorizza il personaggio associato alla battuta che viene passato per parametro Character.

3.15.5.4.4 setSoundFx(SoundFx soundFx)

- **Accesso:** public;

- **Tipo di ritorno:** void;

- **Argomenti:**

- soundFx: l'effetto sonoro da assegnare alla battuta.

- **Descrizione:** Memorizza un effetto sonoro per la battuta che viene passato per parametro SoundFx.

3.15.5.4.5 getText()

- **Accesso:** public;

- **Tipo di ritorno:** String;

- **Descrizione:** Restituisce il testo della battuta.

3.15.5.4.6 getEmotion()

- **Accesso:** public;

- **Tipo di ritorno:** String;

- **Descrizione:** Restituisce l'emozione associata alla battuta.

3.15.5.4.7 getCharacter()

- **Accesso:** public;
- **Tipo di ritorno:** Character;
- **Descrizione:** Restituisce il personaggio associato alla battuta.

3.15.5.4.8 getAudioPath()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Metodo che ritorna il percorso (della memoria del dispositivo) del file audio ottenuto dalla sintesi vocale della battuta.

3.15.5.4.9 setAudioPath(String path)

- **Accesso:** public;
- **Tipo di ritorno:** void; **Argomenti:**
 - path: il path dove salvare la sintesi vocale.
- **Descrizione:** Metodo utile per impostare l'attributo audioStatus che indica se è già stato associato un file audio alla battuta che corrisponde alla sua sintesi vocale.

3.15.5.4.10 getAudioStatus()

- **Accesso:** public;
- **Tipo di ritorno:** boolean;
- **Descrizione:** Metodo utile per conoscere se una battuta ha già associato un file audio corrispondente alla sua sintesi vocale.

3.15.5.4.11 setAudioStatus(boolean status)

- **Accesso:** public;
- **Tipo di ritorno:** void; **Argomenti:**
 - status: il valore booleano da assegnare al campo audioStatus.
- **Descrizione:** Metodo utile per impostare il percorso del file audio corrispondente alla sintesi vocale della battuta.

3.15.5.4.12 synthesizeAudio(Context context, String path)

- **Accesso:** public;
- **Tipo di ritorno:** void; **Argomenti:**
 - context: contesto della view;
 - path: il path dove salvare la sintesi vocale.
- **Descrizione:** Metodo che crea il file .wav ottenuto dalla sintesi vocale della battuta. Il file viene salvato nella memoria del dispositivo.

3.15.5.4.13 getDuration()

- **Accesso:** public;
- **Tipo di ritorno:** long;
- **Descrizione:** Metodo che ritorna la durata in millisecondi di una battuta.

3.15.6 SpeechImpl (class)

SpeechImpl
<pre> - character : Character - emotionID : String - soundFx : SoundFx - text : String - audioPath : String - audioStatus : boolean - SpeechImpl(builder : SpeechBuilder) + setText(text : String) : void + setEmotion(emotionID : String) : void + setCharacter(character : Character) : void + setSoundFx(soundFx : SoundFx) : void + getText() : String + getEmotion() : String + getCharacter() : Character + getEmotions(context : Context) : ArrayAdapter<String> + getVoices(context : Context) : ArrayAdapter<String> + getPath() : String + getAudioStatus() : boolean + getDuration() : long + setAudioPath(audioPath : String) : void + setAudioStatus(audioStatus : boolean) : void + synthesizeAudio(context : Context, path : String) : void </pre>

Figura 97: Classe Sivodim::Drama::Model::Chapter::SpeechImpl

3.15.6.1 Descrizione

Classe che implementa la rappresentazione di un oggetto di tipo battuta. Un oggetto battuta contiene un riferimento ad un oggetto di tipo personaggio.

3.15.6.2 Utilizzo

Viene usata per implementare l'interfaccia del tipo battuta.

3.15.6.3 Classi ereditate

- Sivodim::Drama::Model::Chapter::Speech.

3.15.6.4 Attributi

3.15.6.4.1 text

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Memorizza il testo della battuta.

3.15.6.4.2 emotionID

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Memorizza una String che identifica l'emozione associata alla battuta.

3.15.6.4.3 Character

- **Accesso:** private;
- **Tipo:** Character;
- **Descrizione:** Memorizza un riferimento al personaggio associato alla battuta.

3.15.6.4.4 soundFx

- **Accesso:** private;
- **Tipo:** SoundFx;
- **Descrizione:** Riferimento ad un effetto sonoro che può facoltativamente essere associato alla battuta.

3.15.6.4.5 audioPath

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Campo contenente il percorso nel filesystem in cui vengono salvati i file audio generati a partire dalle battute.

3.15.6.4.6 audioStatus

- **Accesso:** private;
- **Tipo:** boolean;
- **Descrizione:** Campo che indica se la battuta è stata già sintetizzata.

3.15.6.5 Metodi

3.15.6.5.1 SpeechImpl(SpeechBuilder builder)

- **Accesso:** private;
- **Argomenti:**
 - builder: un builder per Speech.
- **Descrizione:** Costruttore privato che costruisce un oggetto Speech a partire dal builder SpeechBuilder.

3.15.6.5.2 setText(String text)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - text: il testo da assingare alla battuta.
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia Speech memorizzando in this.text la String passata per parametro.

3.15.6.5.3 setEmotion(String emotionID)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - emotionID: il nome che identifica l'emozione da assegnare alla battuta.
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia Speech memorizzando in this.emotion la String passata per parametro.

3.15.6.5.4 setCharacter(Character character)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - character: il personaggio da associare alla battuta.
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia Speech memorizzando in this.character il Character passato per parametro.

3.15.6.5.5 setSoundFx(SoundFx soundFx)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - soundFx: l'effetto sonoro da associare alla battuta.
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia Speech memorizzando in this.soundFx il SoundFx passato per parametro.

3.15.6.5.6 getText()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia Chapter restituendo this.text.

3.15.6.5.7 getEmotion()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia Chapter restituendo this.emotion.

3.15.6.5.8 getCharacter()

- **Accesso:** public;
- **Tipo di ritorno:** Character;
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia Chapter restituendo this.character.

3.15.6.5.9 getAudioPath()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Metodo che ritorna il path contenente il file audio sintetizzato della battuta.

3.15.6.5.10 getAudioStatus()

- **Accesso:** public;
- **Tipo di ritorno:** boolean;
- **Descrizione:** Metodo che ritorna true se l'oggetto Speech è già stato sintetizzato.

3.15.6.5.11 getDuration()

- **Accesso:** public;
- **Tipo di ritorno:** long;
- **Descrizione:** Metodo che ritorna la durata di una battuta in millisecondi.

3.15.6.5.12 setAudioStatus(boolean status)

- **Accesso:** public;
- **Tipo di ritorno:** void; **Argomenti:**
 - status: il valore booleano da assegnare al campo audioStatus.
- **Descrizione:** Metodo che setta il campo dato audioStatus a seconda se l'oggetto Speech è già stato sintetizzato.

3.15.6.5.13 setAudioPath(String path)

- **Accesso:** public;
- **Tipo di ritorno:** void; **Argomenti:**
 - path: il path da assegnare.
- **Descrizione:** Metodo che setta il campo dato audioPath.

3.15.6.5.14 getEmotions(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** ArrayAdapter<String>;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Metodo statico che restituisce un ArrayAdapter di String con l'elenco delle emozioni associabili alle battute. Utilizza la componente Sivodim:Libraries per ottenere le informazioni.

3.15.6.5.15 getVoices(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** ArrayAdapter<String>;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Metodo statico che restituisce un ArrayAdapter di String con l'elenco dei nomi identificativi delle voci. Utilizza la componente Sivodim::Libraries per ottenere le informazioni.

3.15.6.5.16 synthesizeAudio(Context context, String path)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il contesto della view;
 - context: il path dove sintetizzare le battute;
- **Descrizione:** Metodo che setta il campo dato audioStatus a seconda se l'oggetto Speech è già stato sintetizzato.

3.15.7 SpeechBuilder (class)

SpeechBuilder
<ul style="list-style-type: none"> - characterB : Character - emotionIDB : String - soundFxB : SoundFx - textB : String - audioPathB : String - audioStatusB : boolean
<ul style="list-style-type: none"> + SpeechBuilder() + setText(text : String) : SpeechBuilder + setEmotion(emotionID : String) : SpeechBuilder + setCharacter(character : Character) : SpeechBuilder + setSoundFX(soundFx : SoundFx) : SpeechBuilder + build() : SpeechImpl + setAudioPath(audioPath : String) : void + setAudioStatus(audioStatus : int) : boolean

Figura 98: Classe Sivodim::Drama::Model::Chapter::SpeechBuilder

3.15.7.1 Descrizione

Classe che implementa la rappresentazione di un oggetto extitBuilder.

3.15.7.2 Utilizzo

Viene usata per implementare l'interfaccia di tipo SpeechBuilder. La creazione di un oggetto di tipo battuta è infatti compito della classe SpeechBuilder strutturata secondo il extitdesign pattern Builder.

3.15.7.3 Attributi

3.15.7.3.1 audioPathB

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Variabile in cui viene mantenuto il path dove salvare la sintetizzazione di una battuta.

3.15.7.3.2 audioStatusB

- **Accesso:** private;
- **Tipo:** boolean;
- **Descrizione:** Variabile settata a true se è già stata effettuata la sintetizzazione della battuta.

3.15.7.3.3 textB

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Variabile in cui viene mantenuto il testo da passare alla Speech al momento della costruzione.

3.15.7.3.4 emotionIDB

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Variabile in cui viene mantenuta l'emozione da dare alla Speech al momento della costruzione.

3.15.7.3.5 characterB

- **Accesso:** private;
- **Tipo:** Character;
- **Descrizione:** Il personaggio da dare alla battuta al momento della creazione.

3.15.7.3.6 soundFxB

- **Accesso:** private;
- **Tipo:** SoundFx;
- **Descrizione:** Variabile in cui viene mantenuto l'eventuale effetto sonoro da associare alla Speech al momento della creazione.

3.15.7.4 Metodi

3.15.7.4.1 SpeechBuilder()

- **Accesso:** public;
- **Descrizione:** Costruttore di *default*.

3.15.7.4.2 setText(String text)

- **Accesso:** public;
- **Tipo di ritorno:** SpeechBuilder;
- **Argomenti:**
 - text: il testo con cui costruire la Speech.
- **Descrizione:** Memorizza il testo da passare alla Speech e ritorna this.

3.15.7.4.3 setEmotion(String emotionID)

- **Accesso:** public;
- **Tipo di ritorno:** SpeechBuilder;
- **Argomenti:**
 - emotionID: la String che identifica l'emozione da passare alla Speech al momento della costruzione.
- **Descrizione:** Memorizza la String che identifica l'emozione che verrà assegnata alla Speech e ritorna this.

3.15.7.4.4 setCharacter(Character character)

- **Accesso:** public;
- **Tipo di ritorno:** SpeechBuilder;
- **Argomenti:**
 - character: riferimento al Character che verrà associato alla Speech al momento della costruzione.
- **Descrizione:** Memorizza il riferimento al Character e restituisce this.

3.15.7.4.5 setSoundFX(SoundFx soundFx)

- **Accesso:** public;
- **Tipo di ritorno:** SpeechBuilder;
- **Argomenti:**
 - soundFx: il SoundFx che viene assegnato alla Speech al momento della creazione.
- **Descrizione:** Memorizza il riferimento al SoundFx e restituisce this.

3.15.7.4.6 build()

- **Accesso:** public;
- **Tipo di ritorno:** SpeechImpl;
- **Descrizione:** Costruisce un oggetto SpeechImpl sulla base dei dati memorizzati negli attributi.

3.15.7.4.7 setAudioPath(String path)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - path: il path dove salvare la sintetizzazione.
- **Descrizione:** Setta il campo dato audioPath.

3.15.7.4.8 setAudioStatus(boolean status)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - status: settato a true se la battuta è già stata sintetizzata.
- **Descrizione:** Setta il campo dato audioStatus.

3.16 Sivodim::Drama::Model::Character

3.16.1 Informazioni sul package

Di seguito viene illustrata, tramite un diagramma delle classi, la visione di insieme del package Drama::Model::Character contenente tutte le classi necessarie per la gestione dei personaggi.

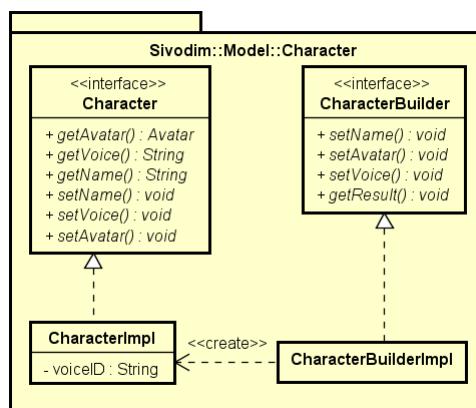


Figura 99: Componente Sivodim::Drama::Model::Character

3.16.2 Character (interface)

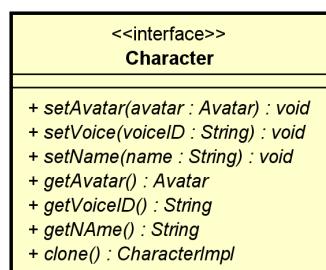


Figura 100: Classe Sivodim::Drama::Model::Character::Character

3.16.2.1 Descrizione

Interfaccia che definisce il contratto attraverso cui implementare un oggetto di tipo personaggio.

3.16.2.2 Utilizzo

Viene usata per definire il contratto da rispettare per l'implementazione di un oggetto di tipo personaggio.

3.16.2.3 Classi figlie

- Sivodim::Drama::Model::Character::CharacterImpl.

3.16.2.4 Metodi

3.16.2.4.1 clone()

- **Accesso:** public;
- **Tipo di ritorno:** Character;
- **Descrizione:** Restituisce una copia profonda dell'oggetto di tipo Character su cui è invocato.

3.16.2.4.2 getAvatar()

- **Accesso:** public;
- **Tipo di ritorno:** Avatar;
- **Descrizione:** Restituisce un riferimento all'oggetto di tipo Avatar che rappresenta l'avatar del personaggio;
- **Note:** Se il personaggio non ha un avatar restituisce null.

3.16.2.4.3 getName()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Restituisce il nome del personaggio.

3.16.2.4.4 getVoiceld()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Restituisce la stringa che identifica la voce associata al personaggio.

3.16.2.4.5 setAvatar(Avatar avatar)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - avatar: l'avatar da assegnare al personaggio.
- **Descrizione:** Prende per parametro un riferimento ad Avatar e lo setta come avatar del personaggio.

3.16.2.4.6 setName(String name)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - name: il nome da assegnare al personaggio.
- **Descrizione:** Prende per parametro una String e la imposta come nome del personaggio;
- **Note:** Il nome del personaggio deve essere univoco per identificarlo.

3.16.2.4.7 setVoice(String voiceID)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - voiceID: il nome che identifica una voce.
- **Descrizione:** Prende per parametro una String che identifica una voce e la associa al personaggio.

3.16.3 CharacterImpl (class)

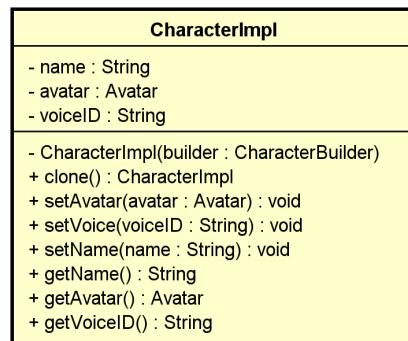


Figura 101: Classe Sivodim::Drama::Model::Character::CharacterImpl

3.16.3.1 Descrizione

Classe che implementa la rappresentazione di un oggetto di tipo personaggio. La classe ha un riferimento a un oggetto di tipo Avatar.

3.16.3.2 Utilizzo

Viene usata per implementare l'interfaccia del tipo personaggio.

3.16.3.3 Classi ereditate

- Sivodim::Drama::Model::Character::Character.

3.16.3.4 Attributi

3.16.3.4.1 avatar

- **Accesso:** private;
- **Tipo:** Avatar;
- **Descrizione:** Mantiene un riferimento ad un oggetto di tipo Avatar che rappresenta l'avatar del personaggio.

3.16.3.4.2 name

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Memorizza il nome del Character.

3.16.3.4.3 voicelD

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Memorizza la String identificativa della voce del personaggio.

3.16.3.4.4 name

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Stringa di testo che rappresenta il nome associato a un personaggio.

3.16.3.4.5 avatar

- **Accesso:** private;
- **Tipo:** Avatar;
- **Descrizione:** File immagine di tipo Avatar che rappresenta l'*avatar* associato a un personaggio.

3.16.3.4.6 voicelD

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Stringa che contiene l'identificativo del *file* contenente i parametri che caratterizzano la voce associata al personaggio.

3.16.3.5 Metodi

3.16.3.5.1 CharacterImpl(CharacterBuilder builder)

- **Accesso:** public;
- **Argomenti:**
 - builder: un builder per il personaggio.
- **Descrizione:** Costruttore della classe. Inizializza i campi dati sulla base di un *builder* di tipo CharacterBuilder.

3.16.3.5.2 clone()

- **Accesso:** public;
- **Tipo di ritorno:** Character;
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia Character. Effettua una copia profonda del Character d'invocazione e ne restituisce un riferimento.

3.16.3.5.3 setAvatar(Avatar avatar)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - avatar: l'avatar da assegnare al personaggio.
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia Character memorizzando un riferimento ad Avatar passato per parametro come $avatar_G$ per il personaggio.

3.16.3.5.4 setVoice(String voiceID)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - voiceID: il nome che identifica la voce da assegnare al personaggio.
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia Character. Memorizza una String, passata per parametro, che identifica la voce del personaggio.

3.16.3.5.5 setName(String name)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - name: il nome da assegnare al personaggio.
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia Character. Memorizza il nome del personaggio, che viene passato attraverso un parametro di tipo String.

3.16.3.5.6 getName()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia Character restituendo this.name.

3.16.3.5.7 getAvatar()

- **Accesso:** public;
- **Tipo di ritorno:** Avatar;
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia Character restituendo this.avatar.

3.16.3.5.8 getVoiceId()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia Character restituendo this.voiceID.

3.16.4 CharacterBuilder (class)

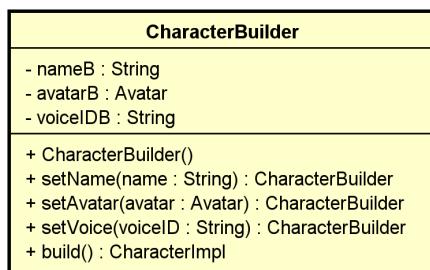


Figura 102: Classe Sivodim::Drama::Model::Character::CharacterBuilder

3.16.4.1 Descrizione

Classe statica interna a CharacterImpl che implementa la rappresentazione di un oggetto *Builder*.

3.16.4.2 Utilizzo

Viene usata per implementare l'interfaccia di tipo CharacterBuilder. La creazione di un oggetto di tipo personaggio è infatti compito della classe CharacterBuilder strutturata secondo il *design pattern Builder*.

3.16.4.3 Classi ereditate

- Sivodim::Drama::Model::Character::CharacterBuilder.

3.16.4.4 Metodi

3.16.4.4.1 CharacterBuilder()

- **Accesso:** public;
- **Descrizione:** Costruttore di default.

3.16.4.4.2 setName(String name)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - name: il nome da assegnare al personaggio.
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia CharacterBuilder impostando this.nameB al parametro String ricevuto e ritornando this.

3.16.4.4.3 setAvatar(Avatar avatar)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - avatar: l'Avatar da assegnare al personaggio.
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia CharacterBuilder impostando this.avatarB al parametro Avatar ricevuto e ritornando this.

3.16.4.4.4 setVoice(String voiceID)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - voiceID: il nome che identifica una voce.
- **Descrizione:** Implementa il metodo dichiarato nell'interfaccia CharacterBuilder impostando this.voiceIDB al parametro String ricevuto e ritornando this.

3.16.5 CharacterContainer (class)

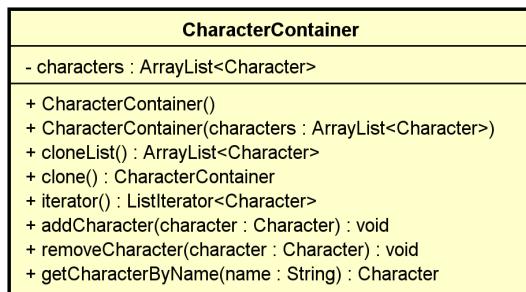


Figura 103: Classe Sivodim::Drama::Model::Character::CharacterContainer

3.16.5.1 Descrizione

Classe che implementa il contenitore dei personaggi relativi a uno sceneggiato di tipo Screenplay. Contiene una lista di personaggi di tipo Character.

3.16.5.2 Utilizzo

Necessaria per importare i personaggi da uno sceneggiato a un altro.

3.16.5.3 Attributi

3.16.5.3.1 characters

- **Accesso:** private;
- **Tipo:** ArrayList<Character>;
- **Descrizione:** Memorizza i personaggi che il CharacterContainer gestisce.

3.16.5.4 Metodi

3.16.5.4.1 CharacterContainer()

- **Accesso:** public;
- **Descrizione:** Costruttore di default.

3.16.5.4.2 CharacterContainer(ArrayList<Character> characters)

- **Accesso:** public;
- **Argomenti:**
 - characters: ArrayList contenente personaggi.
- **Descrizione:** Costruttore che inizializza la lista dei personaggi.

3.16.5.4.3 cloneList()

- **Accesso:** public;
- **Tipo di ritorno:** ArrayList<Character>;
- **Descrizione:** Restituisce una copia profonda del contenitore (ArrayList<Character>) di personaggi.

3.16.5.4.4 clone()

- **Accesso:** public;
- **Tipo di ritorno:** CharacterContainer;
- **Descrizione:** Restituisce una copia profonda del CharacterContainer. Utilizza cloneList().

3.16.5.4.5 iterator()

- **Accesso:** public;
- **Tipo di ritorno:** ListIterator<Character>;
- **Descrizione:** Restituisce un iteratore di Character che consente di iterare sui personaggi contenuti nel CharacterContainer.

3.16.5.4.6 addCharacter(Character character)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - character: il personaggio da aggiungere al CharacterContainer.
- **Descrizione:** Aggiunge al contenitore di CharacterContainer un riferimento al personaggio indicato.

3.16.5.4.7 removeCharacter(Character character)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - character: il Character da rimuovere da CharacterContainer.
- **Descrizione:** Rimuove dal contenitore di CharacterContainer il riferimento al personaggio indicato.

3.16.5.4.8 getCharacterByName(String name)

- **Accesso:** public;
- **Tipo di ritorno:** Character;
- **Argomenti:**
 - name: il nome del personaggio da trovare.
- **Descrizione:** Restituisce il riferimento al Character contenuto nel CharacterContainer che ha il nome passato per parametro;
- **Note:** Restituisce null se il personaggio non è presente.

3.17 Sivodim::Drama::Model::Screenplay

3.17.1 Informazioni sul package

Di seguito viene illustrata, tramite un diagramma delle classi, la visione di insieme del package Drama::Model::Screenplay contenente tutte le classi necessarie per la gestione di sceneggiati e la loro conversione (in audio e video) attraverso la libreria FFmpeg.

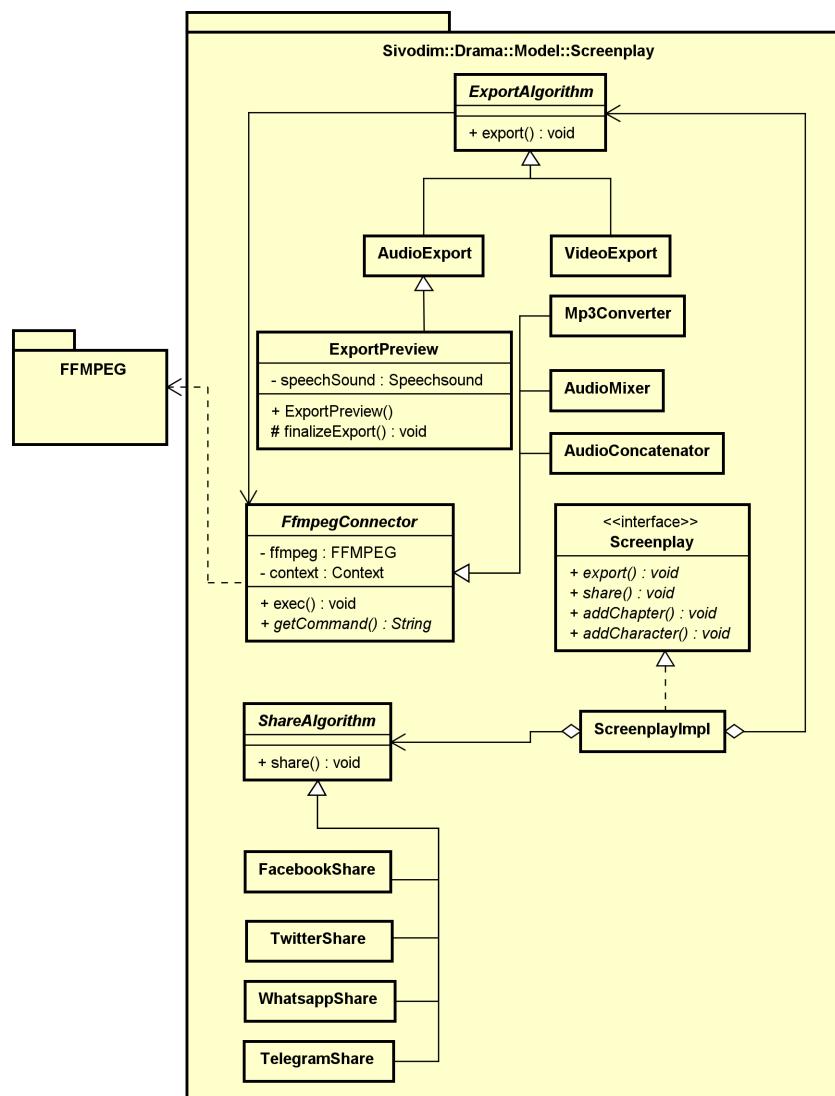


Figura 104: Componente Sivodim::Drama::Model::Screenplay

3.17.2 Screenplay (interface)

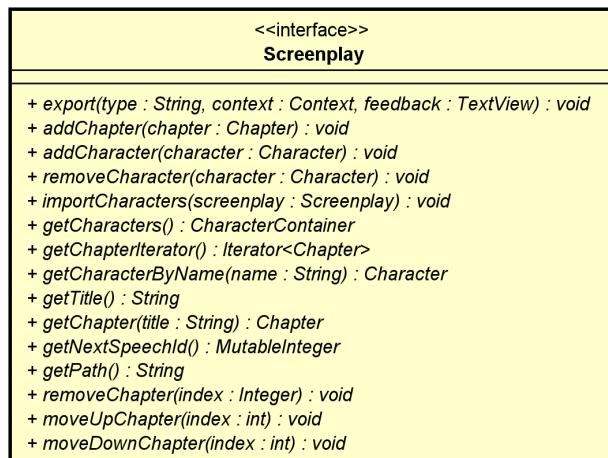


Figura 105: Classe Sivodim::Drama::Model::Screenplay::Screenplay

3.17.2.1 Descrizione

Interfaccia che determina il contratto di un oggetto di tipo sceneggiato e che mette a disposizione i metodi di esportazione e condivisione dello sceneggiato.

3.17.2.2 Utilizzo

Viene usata per determinare il contratto di un oggetto di tipo sceneggiato.

3.17.2.3 Classi figlie

- Sivodim::Drama::Model::Screenplay::ScreenplayImpl.

3.17.2.4 Metodi

3.17.2.4.1 addChapter(Chapter chapter)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - chapter: il capitolo da aggiungere.
- **Descrizione:** Prende un oggetto di tipo Chapter (capitolo) esistente e lo inserisce nello Screenplay.

3.17.2.4.2 addCharacter(Character character)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - character: il personaggio da aggiungere.
- **Descrizione:** Prende un riferimento a Character e lo inserisce nel CharacterContainer dello Screenplay.

3.17.2.4.3 export(String type, Context context, TextView feedback)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - type: stringa relativa alla tipologia di esportazione;
 - context: il contesto della view;
 - feedback: una TextView che mostra l'avanzamento dell'esportazione.
- **Descrizione:** Istanzia un ExportAlgorithm e lo usa per esportare lo sceneggiato, prendendo per parametro le informazioni sul formato di esportazione.

3.17.2.4.4 getChapter(String title)

- **Accesso:** public;
- **Tipo di ritorno:** Chapter;
- **Argomenti:**
 - title: il titolo del capitolo da cercare.
- **Descrizione:** Ritorna un Chapter contenuto nello Screenplay cercandolo in base al titolo, che viene richiesto come parametro;
- **Note:** Il titolo di un Chapter deve essere univoco.

3.17.2.4.5 getChapterIterator()

- **Accesso:** public;
- **Tipo di ritorno:** Iterator<Chapter>;
- **Descrizione:** Restituisce un iteratore di Chapter che consente di scorrere fra i capitoli presenti nello Screenplay;
- **Note:** Il contenitore presente nell'implementazione di questa interfaccia deve essere iterabile.

3.17.2.4.6 getCharacterByName(String name)

- **Accesso:** public;
- **Tipo di ritorno:** Character;
- **Argomenti:**
 - name: il nome del personaggio da cercare.
- **Descrizione:** Restituisce un riferimento all'oggetto Character contenuto nello Screenplay avente lo stesso nome del parametro passato al metodo;
- **Note:** Il nome dei personaggi deve essere univoco all'interno di uno stesso Screenplay.

3.17.2.4.7 getCharacters()

- **Accesso:** public;
- **Tipo di ritorno:** CharacterContainer;
- **Descrizione:** Restituisce un riferimento al CharacterContainer dello Screenplay, ossia al contenitore dei personaggi dello sceneggiato;
- **Note:** I personaggi dello Screenplay vengono memorizzati all'interno di un CharacterContainer, che è mantenuto dallo Screenplay, per facilitarne lo scambio fra oggetti.

3.17.2.4.8 getTitle()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Restituisce il titolo dello Screenplay;
- **Note:** Il titolo dello Screenplay deve identificarlo e pertanto deve essere univoco.

3.17.2.4.9 importCharacters(Screenplay screenplay)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - screenplay: lo Screenplay da cui importare i personaggi.
- **Descrizione:** Importa nello Screenplay una copia dei personaggi di un altro Screenplay;
- **Note:** Deve essere effettuata una copia profonda degli oggetti.

3.17.2.4.10 removeCharacter(Character character)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - character: il personaggio da togliere.
- **Descrizione:** Toglie un personaggio dallo Screenplay, prendendo per parametro un riferimento all'oggetto Character da togliere.

3.17.2.4.11 removeChapter(Chapter chapter)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - chapter: il capitolo da togliere.
- **Descrizione:** Toglie un capitolo dallo Screenplay, prendendo per parametro un riferimento all'oggetto Chapter da togliere.

3.17.2.4.12 moveUpChapter(int index)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - index: il capitolo da spostare.
- **Descrizione:** Sposta di una posizione in meno un capitolo dello Screenplay.

3.17.2.4.13 moveDownChapter(int index)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - index: il capitolo da spostare.
- **Descrizione:** Sposta di una posizione in più un capitolo dello Screenplay.

3.17.2.4.14 getPath()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Ritorna il percorso della memoria del dispositivo dell'audio associato allo Screenplay.

3.17.2.4.15 getNextSpeechId()

- **Accesso:** public;
- **Tipo di ritorno:** MutableInteger;
- **Descrizione:** identifica con un id ogni Speech dello sceneggiato.

3.17.3 ScreenplayImpl (class)

ScreenplayImpl
<pre> - title : String - characterContainer : CharacterContainer - chapters : ArrayList<Character> - exportAlgorithm : ExportAlgorithm - nextSpeechId : MutableInteger + ScreenplayImpl(title : String, nextSpeechId : int) + loadScreenplay(screenplayTitle : String, context : Context) : Screenplay + saveScreenplay(screenplay : Screenplay, context : Context) : void + getChapterIterator() : Iterator<Chapter> + export(type : String, context : Context, feedback : TextView) : void + getTitle() : String + getCharacters() : CharacterContainer + importCharacters(screenplay : Screenplay) : void + addCharacter(character : Character) : void + removeCharacter(character : Character) : void + getChapter(title : String) : Chapter + getCharacterByName(name : String) : Character + addChapter(chapter : Chapter) : void + getNextSpeechId() : int + getPath() : String + moveDownChapter(index : int) : void + moveUpChapter(index : int) : void + remove Chapter(index : int) : void </pre>

Figura 106: Classe Sivodim::Drama::Model::Screenplay::ScreenplayImpl

3.17.3.1 Descrizione

Classe che rappresenta l'implementazione di un oggetto di tipo sceneggiato. Ha come attributi una struttura dati contenente un certo numero di oggetti capitolo, una struttura dati contenente un certo numero di oggetti personaggio e un'istanza degli algoritmi di esportazione e condivisione.

3.17.3.2 Utilizzo

Viene utilizzata per istanziare un oggetto di tipo sceneggiato.

3.17.3.3 Classi ereditate

- Sivodim::Drama::Model::Screenplay::Screenplay.

3.17.3.4 Attributi

3.17.3.4.1 chapters

- **Accesso:** private;
- **Tipo:** ArrayList<Chapter>;
- **Descrizione:** Contenitore che mantiene i riferimenti ai capitoli (oggetti che implementano l'interfaccia Chapter) dello Screenplay.

3.17.3.4.2 characterContainer

- **Accesso:** private;
- **Tipo:** CharacterContainer;
- **Descrizione:** Contenitore dei personaggi dello Screenplay.

3.17.3.4.3 exportAlgorithm

- **Accesso:** private;
- **Tipo:** ExportAlgorithm;
- **Descrizione:** Riferimento ad un oggetto ExportAlgorithm che viene utilizzato per effettuare l'esportazione dello sceneggiato.

3.17.3.4.4 nextSpeechId

- **Accesso:** private;
- **Tipo:** MutableInteger;
- **Descrizione:** Identifica ogni battuta con un id.

3.17.3.4.5 title

- **Accesso:** private;
- **Tipo:** String;
- **Descrizione:** Il titolo dello Screenplay. Deve essere univoco per identificare gli Screenplay.

3.17.3.5 Metodi

3.17.3.5.1 ScreenplayImpl(String title, int nextSpeechId)

- **Accesso:** public;
- **Argomenti:**
 - title: il titolo dello Screenplay;
 - nextSpeechId: id che identifica una battuta.
- **Descrizione:** Costruttore dello sceneggiato. Prende per parametro una String con cui inizializza il titolo dello ScreenplayImpl;
- **Note:** Il titolo deve essere univoco per permettere l'identificazione dello sceneggiato. Deve pertanto essere effettuato un controllo sulla disponibilità del nome. In caso contrario viene aggiunto un numero progressivo al termine della stringa.

3.17.3.5.2 loadScreenplay(String screenplayTitle, Context context)

- **Accesso:** public;
- **Tipo di ritorno:** Screenplay;
- **Argomenti:**
 - screenplayTitle: il titolo dello Screenplay;
 - context: il contesto della view.
- **Descrizione:** Metodo statico che prende per parametro il titolo dello sceneggiato e un Context. Utilizza la classe XMLParser per caricare dalla memoria lo sceneggiato avente il nome passato per parametro;
- **Note:** Lo sceneggiato deve essere salvato nella cartella di default per l'app.

3.17.3.5.3 saveScreenplay(Screenplay screenplay, Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - screenplay: lo Screenplay da salvare;
 - context: il contesto della view.
- **Descrizione:** Metodo statico che prende per parametro un riferimento ad un oggetto di tipo Screenplay e un Context. Utilizza la classe XMLParser per salvare in memoria lo sceneggiato nella cartella di default.

3.17.3.5.4 addChapter(Chapter chapter)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - chapter: il capitolo da aggiungere.
- **Descrizione:** Prende un oggetto di tipo Chapter (capitolo) esistente e lo inserisce nello Screenplay.

3.17.3.5.5 addCharacter(Character character)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - character: il personaggio da aggiungere.
- **Descrizione:** Prende un riferimento a Character e lo inserisce nel CharacterContainer dello Screenplay.

3.17.3.5.6 export(String type, Context context, TextView feedback)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - type: stringa relativa alla tipologia di esportazione;
 - context: il contesto della view;
 - feedback: una TextView che mostra l'avanzamento dell'esportazione.
- **Descrizione:** Istanzia un ExportAlgorithm e lo usa per esportare lo sceneggiato, prendendo per parametro le informazioni sul formato di esportazione.

3.17.3.5.7 getChapter(String title)

- **Accesso:** public;
- **Tipo di ritorno:** Chapter;
- **Argomenti:**
 - title: il titolo del capitolo da cercare.
- **Descrizione:** Ritorna un Chapter contenuto nello Screenplay cercandolo in base al titolo, che viene richiesto come parametro;
- **Note:** Il titolo di un Chapter deve essere univoco.

3.17.3.5.8 getChapterIterator()

- **Accesso:** public;
- **Tipo di ritorno:** Iterator<Chapter>;
- **Descrizione:** Restituisce un iteratore di Chapter che consente di scorrere fra i capitoli presenti nello Screenplay;
- **Note:** Il contenitore presente nell'implementazione di questa interfaccia deve essere iterabile.

3.17.3.5.9 getCharacterByName(String name)

- **Accesso:** public;
- **Tipo di ritorno:** Character;
- **Argomenti:**
 - name: il nome del personaggio da cercare.
- **Descrizione:** Restituisce un riferimento all'oggetto Character contenuto nello Screenplay avente lo stesso nome del parametro passato al metodo;
- **Note:** Il nome dei personaggi deve essere univoco all'interno di uno stesso Screenplay.

3.17.3.5.10 getCharacters()

- **Accesso:** public;
- **Tipo di ritorno:** CharacterContainer;
- **Descrizione:** Restituisce un riferimento al CharacterContainer dello Screenplay, ossia al contenitore dei personaggi dello sceneggiato;
- **Note:** I personaggi dello Screenplay vengono memorizzati all'interno di un CharacterContainer, che è mantenuto dallo Screenplay, per facilitarne lo scambio fra oggetti.

3.17.3.5.11 getTitle()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Restituisce il titolo dello Screenplay;
- **Note:** Il titolo dello Screenplay deve identificarlo e pertanto deve essere univoco.

3.17.3.5.12 importCharacters(Screenplay screenplay)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - screenplay: lo Screenplay da cui importare i personaggi.
- **Descrizione:** Importa nello Screenplay una copia dei personaggi di un altro Screenplay;
- **Note:** Deve essere effettuata una copia profonda degli oggetti.

3.17.3.5.13 removeCharacter(Character character)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - character: il personaggio da togliere.
- **Descrizione:** Toglie un personaggio dallo Screenplay, prendendo per parametro un riferimento all'oggetto Character da togliere.

3.17.3.5.14 removeChapter(Chapter chapter)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - chapter: il capitolo da togliere.
- **Descrizione:** Toglie un capitolo dallo Screenplay, prendendo per parametro un riferimento all'oggetto Chapter da togliere.

3.17.3.5.15 moveUpChapter(int index)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - index: il capitolo da spostare.
- **Descrizione:** Sposta di una posizione in meno un capitolo dello Screenplay.

3.17.3.5.16 moveDownChapter(int index)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - index: il capitolo da spostare.
- **Descrizione:** Sposta di una posizione in più un capitolo dello Screenplay.

3.17.3.5.17 getPath()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Ritorna il percorso della memoria del dispositivo dell'audio associato allo Screenplay.

3.17.3.5.18 getNextSpeechId()

- **Accesso:** public;
- **Tipo di ritorno:** MutableInteger;
- **Descrizione:** identifica con un id ogni Speech dello sceneggiato.

3.17.4 FfmpegConnector (abstract)

<i>FfmpegConnector</i>	
- ffmpeg : FFmpeg	
- context : Context	
- errors : boolean	
- finish : boolean	
+ FfmpegConnector(context : Context)	
+ exec() : void	
+ getCommand() : void	
+ isErrors() : boolean	

Figura 107: Classe Sivodim::Drama::Presenter::FfmpegConnector

3.17.4.1 Descrizione

Classe astratta che si relaziona con FFMPEG, passandone i comandi e ricevendo i messaggi generati dall'esecuzione.

3.17.4.2 Utilizzo

Viene utilizzata per comunicare con FFMPEG per passare comandi.

3.17.4.3 Classi figlie

- Sivodim::Drama::Model::Screenplay::Mp3Converter;
- Sivodim::Drama::Model::Screenplay::AudioConcatenator;
- Sivodim::Drama::Model::Screenplay::AudioMixer.

3.17.4.4 Attributi

3.17.4.4.1 ffmpeg

- **Accesso:** private;
- **Tipo:** FFMPEG;
- **Descrizione:** Handler a FFMPEG.

3.17.4.4.2 context

- **Accesso:** private;
- **Tipo:** Context;
- **Descrizione:** Contesto dell'applicazione Android.

3.17.4.4.3 errors

- **Accesso:** private;
- **Tipo:** boolean;
- **Descrizione:** Flag che segnala la presenza di errori avvenuti durante l'esecuzione di FFMPEG.

3.17.4.4.4 finish

- **Accesso:** private;
- **Tipo:** boolean;
- **Descrizione:** Flag che segnala se l'esecuzione di FFMPEG è giunta a termine.

3.17.4.5 Metodi

3.17.4.5.1 FfmpegConnector(Context context)

- **Accesso:** public;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Costruisce un oggetto della classe e inizializza il modulo ffmpeg nel dispositivo.

3.17.4.5.2 exec()

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Descrizione:** Fa eseguire il comando ottenuto tramite il metodo this.getCommand() alla componente ffmpeg precedentemente avviata nel dispositivo.

3.17.4.5.3 getCommand()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Metodo astratto che deve essere implementato nelle sottoclassi per fornire il comando da far eseguire alla componente ffmpeg.

3.17.5 ExportAlgorithm (abstract)

<i>ExportAlgorithm</i>
screenplay : Screenplay
+ export(context : Context) : void
+ setScreenplay(screenplay : Screenplay) : void
+ ExportAlgorithm()

Figura 108: Classe Sivodim::Drama::Model::Screenplay::ExportAlgorithm

3.17.5.1 Descrizione

Classe astratta che racchiude gli algoritmi di esportazione.

3.17.5.2 Utilizzo

Viene definito il contratto di un algoritmo di esportazione.

3.17.5.3 Classi figlie

- Sivodim::Drama::Model::Screenplay::AudioExport;
- Sivodim::Drama::Model::Screenplay::VideoExport.

3.17.5.4 Attributi

3.17.5.4.1 screenplay

- **Accesso:** protected;
- **Tipo:** Screenplay;
- **Descrizione:** Riferimento allo Screenplay da esportare.

3.17.5.5 Metodi

3.17.5.5.1 export(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il context della view.
- **Descrizione:** Metodo astratto che deve essere definito nelle sottoclassi per implementare l'esportazione.

3.17.5.5.2 ExportAlgorithm()

- **Accesso:** public;
- **Descrizione:** Costruttore di default.

3.17.5.5.3 setScreenplay(Screenplay screenplay)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - screenplay: lo Screenplay da esportare.
- **Descrizione:** Imposta lo Screenplay da esportare.

3.17.6 AudioExport (class)

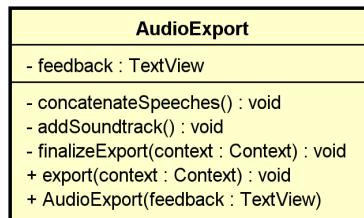


Figura 109: Classe Sivodim::Drama::Model::Screenplay::AudioExport

3.17.6.1 Descrizione

Classe che implementa l'algoritmo di esportazione di un file audio.

3.17.6.2 Utilizzo

Viene usata per esportare nella memoria del dispositivo il file audio ottenuto da uno sceneggiato.

3.17.6.3 Classi ereditate

- Sivodim::Drama::Model::Screenplay::ExportAlgorithm.

3.17.6.4 Classi figlie

- Sivodim::Drama::Model::Screenplay::Exportpreview;

3.17.6.5 Attributi

3.17.6.5.1 feedback

- **Accesso:** private;
- **Tipo:** TextView;
- **Descrizione:** View che mostra l'avanzamento dell'esportazione.

3.17.6.6 Metodi

3.17.6.6.1 AudioExport(TextView feedback)

- **Accesso:** public;
- **Argomenti:**
 - feedback: view che mostra lo stato di avanzamento.
- **Descrizione:** Costruttore di default.

3.17.6.6.2 concatenateSpeeches()

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Descrizione:** Concatena gli SpeechSound ottenuti dalle battute.

3.17.6.6.3 addSoundtrack()

- **Accesso:** private;
- **Tipo di ritorno:** void;
- **Descrizione:** Aggiunge all'audio ottenuto dalla concatenazione l'audio di sottofondo.

3.17.6.6.4 finalizeExport(Context context)

- **Accesso:** private;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il context della view.
- **Descrizione:** Effettua la conversione in MP3 dell'audio prodotto.

3.17.6.6.5 export(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il context della view.
- **Descrizione:** Implementa il metodo astratto ereditato da ExportAlgorithm e utilizza i metodi privati di utilità della classe.

3.17.7 ExportPreview (class)

ExportPreview
- speechSound : Speechsound
+ ExportPreview(textView : TextView, speechSound : Speechsound) # finalizeExport(final context : Context) : void

Figura 110: Classe Sivodim::Drama::Model::Screenplay::ExportPreview

3.17.7.1 Descrizione

Classe che estende AudioExport l'algoritmo di esportazione di un file audio.

3.17.7.2 Utilizzo

Viene usata per creare un'anteprima istantanea delle battute all'interno di un capitolo.

3.17.7.3 Classi ereditate

- Sivodim::Drama::Model::Screenplay::AudioExport.

3.17.7.4 Attributi

3.17.7.4.1 speechSound

- **Accesso:** private;
- **Tipo:** SpeechSound;
- **Descrizione:** Rappresenta un file sonoro in memoria relativo alla battuta.

3.17.7.5 Metodi

3.17.7.5.1 ExportPreview(TextView textView, SpeechSound speechSound)

- **Accesso:** public;
- **Argomenti:**
 - textView: view che mostra lo stato di avanzamento.
 - speechSound: file sonoro di una battuta.
- **Descrizione:** Costruttore di default.

3.17.7.5.2 finalizeExport(Context context)

- **Accesso:** private;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il context della view.
- **Descrizione:** Effettua la conversione in MP3 dell'audio prodotto.

3.17.8 VideoExport (class)

VideoExport
<ul style="list-style-type: none"> - audioExport : AudioForVideoExport - feedback : TextView - MEDIA_TYPE_IMAGE : int - MEDIA_TYPE_VIDEO : int - timePassed : long - exportAudio(context : Context) : void - concatenateChapters(context : Context, chapterExports : ArrayList<File>) : void - finalizeExport() : void + export(context : Context) : void + VideoExport(feedback : TextView) - createOverlay() : void - makeChapterVideo() : void - getOutputMediaFile() : void - getOutputMediaFileUri() : void

Figura 111: Classe Sivodim::Drama::Model::Screenplay::VideoExport

3.17.8.1 Descrizione

Classe che implementa l'algoritmo di esportazione di un file video.

3.17.8.2 Utilizzo

Viene usata per esportare nella memoria del dispositivo il file video ottenuto da uno sceneggiato.

3.17.8.3 Classi ereditate

- Sivodim::Drama::Model::Screenplay::ExportAlgorithm.

3.17.8.4 Attributi

3.17.8.4.1 audioExport

- **Accesso:** private;
- **Tipo:** AudioForVideoExport;
- **Descrizione:** Riferimento ad un oggetto AudioForVideoExport utilizzato per creare l'audio dello sceneggiato.

3.17.8.4.2 feedback

- **Accesso:** private;
- **Tipo:** TextView;
- **Descrizione:** View che mostra l'avanzamento dell'esportazione.

3.17.8.4.3 mediaTypeImage

- **Accesso:** private;
- **Tipo:** int;
- **Descrizione:** campo dati statico.

3.17.8.4.4 mediaTypeVideo

- **Accesso:** private;
- **Tipo:** int;
- **Descrizione:** campo dati statico.

3.17.8.4.5 timePassed

- **Accesso:** private;
- **Tipo:** long;
- **Descrizione:** tempo passato dall'inizio dell'operazione di esportazione.

3.17.8.5 Metodi

3.17.8.5.1 VideoExport(TextView feedback)

- **Accesso:** public;
- **Argomenti:**
 - feedback: view che mostra l'avanzamento dell'esportazione.
- **Descrizione:** Costruttore di default.

3.17.8.5.2 exportAudio(Context context)

- **Accesso:** private;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il context della view.
- **Descrizione:** Crea l'audio dello sceneggiato utilizzando il riferimento a AudioExport.

3.17.8.5.3 concatenateChapters(Context context, ArrayList<File> chapterExporters)

- **Accesso:** private;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il context della view;
 - chaptersExporters: lista dei file video esportati relativi ad ogni capitolo dello sceneggiato.
- **Descrizione:** Crea la parte visiva del video concatenando immagini prese dagli avatar e dagli sfondi dei capitoli.

3.17.8.5.4 finalizeExport()

- **Accesso:** private;
- **Tipo di ritorno:** void;
- **Descrizione:** Effettua la conversione in MP4_G.

3.17.8.5.5 export(Context context)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - context: il context della view.
- **Descrizione:** Implementazione del metodo astratto ereditato dalla classe ExportAlgorithm.

3.17.8.5.6 createOverlay()

- **Accesso:** private;
- **Tipo di ritorno:** void;
- **Descrizione:** Crea un overlay per l'immagine di sfondo.

3.17.8.5.7 makeChapterVideo()

- **Accesso:** private;
- **Tipo di ritorno:** void;
- **Descrizione:** Genera un video per un Chapter.

3.17.8.5.8 getOutputMediaFile()

- **Accesso:** private;
- **Tipo di ritorno:** void;
- **Descrizione:** Metodo statico che crea un file per salvare immagini o video.

3.17.8.5.9 getOutputMediaFileUri()

- **Accesso:** private;
- **Tipo di ritorno:** void;
- **Descrizione:** Metodo statico che crea un file uri per salvare immagini o video.

3.17.9 Mp3Converter (class)

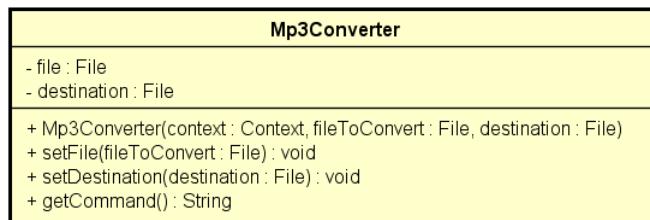


Figura 112: Classe Sivodim::Drama::Model::Screenplay::Mp3Converter

3.17.9.1 Descrizione

Classe concreta che estende FfmpegConnector e ne implementa il metodo astratto getCommand per gestire la conversione audio di file WAVG in MP3G.

3.17.9.2 Utilizzo

Viene utilizzata dalla classe ScreenplayImpl nella fase di esportazione per convertire l'audio in MP3.

3.17.9.3 Classi ereditate

- Sivodim::Drama::Model::Screenplay::FfmpegConnector.

3.17.9.4 Attributi

3.17.9.4.1 file

- **Accesso:** private;
- **Tipo:** File;
- **Descrizione:** Tiene traccia del File da convertire in MP3_G.

3.17.9.4.2 destination

- **Accesso:** private;
- **Tipo:** File;
- **Descrizione:** Il File di destinazione dove viene salvato l'audio convertito in MP3_G.

3.17.9.5 Metodi

3.17.9.5.1 Mp3Converter(Context context, File fileToConvert, File destination)

- **Accesso:** public;
- **Argomenti:**
 - context: il contesto della view;
 - fileToConvert: il File da convertire in MP3_G;
 - destination: il File di destinazione dove viene salvato l'audio.
- **Descrizione:** Costruttore che riceve in input il File da convertire in MP3 e la sua destinazione.

3.17.9.5.2 setFile(File fileToConvert)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - fileToConvert: il File da convertire in MP3.
- **Descrizione:** Metodo setter che reimposta il file da convertire in MP3.

3.17.9.5.3 setDestination(File destination)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - destination: il File in cui verrà salvata la conversione.
- **Descrizione:** Metodo setter che reimposta il File di destinazione.

3.17.9.5.4 getCommand()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Definizione del metodo astratto ereditato da FfmpegConnector, che restituisce il comando da dare a ffmpeg per convertire il file di cui si ha un riferimento in MP3.

3.17.10 AudioConcatenator (class)

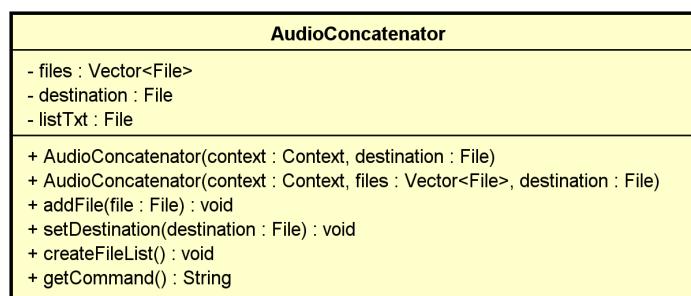


Figura 113: Classe Sivodim::Drama::Model::Screenplay::AudioConcatenator

3.17.10.1 Descrizione

Classe concreta che estende FfmpegConnector e ne implementa il metodo astratto getCommand per gestire la concatenazione di file audio di tipo WAV_G.

3.17.10.2 Utilizzo

Viene utilizzata dalla classe ScreenplayImpl nella fase di esportazione per concatenare i file audio generati a partire dalle battute.

3.17.10.3 Classi ereditate

- Sivodim::Drama::Model::Screenplay::FfmpegConnector.

3.17.10.4 Attributi

3.17.10.4.1 files

- **Accesso:** private;
- **Tipo:** Vector<File>;
- **Descrizione:** Vector di File contenente la lista dei file audio da concatenare.

3.17.10.4.2 destination

- **Accesso:** public;
- **Tipo:** File;
- **Descrizione:** File di destinazione in cui salvare l'audio finale.

3.17.10.4.3 listTxt

- **Accesso:** private;
- **Tipo:** File;
- **Descrizione:** File di utilità, necessario a FFmpeg per effettuare l'operazione di concatenazione.

3.17.10.5 Metodi

3.17.10.5.1 AudioConcatenator(Context context, File destination)

- **Accesso:** public;
- **Argomenti:**
 - context: il contesto della view.
- **Descrizione:** Costruttore che prende il contesto e il percorso di destinazione della concatenazione, inizializzando a null i la lista di file da concatenare.

3.17.10.5.2 AudioConcatenator(Context context, Vector<File> files, File destination)

- **Accesso:** public;
- **Argomenti:**
 - context: il contesto della view;
 - files: l'insieme di file che devono essere concatenati;
 - destination: il file di destinazione in cui verrà salvata la concatenazione.
- **Descrizione:** Costruttore che inizializza l'elenco di file che devono essere concatenati.

3.17.10.5.3 addFile(File file)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - file: il File che deve essere aggiunto alla coda di concatenazione.
- **Descrizione:** Aggiunge un file alla coda di concatenazione.

3.17.10.5.4 setDestination(File destination)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - destination: il file di destinazione.
- **Descrizione:** Riassegna il file di destinazione con il parametro fornito.

3.17.10.5.5 createFileList()

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Descrizione:** Crea un file testuale contenente l'elenco dei percorsi per i file audio che devono essere concatenati, nel rispetto della sintassi usata da ffmpeg.

3.17.10.5.6 getCommand()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Definizione del metodo astratto ereditato da FfmpegConnector. Restituisce una String contenente il comando per ffmpeg per la concatenazione dei file audio presenti nello stato.

3.17.11 AudioMixer (class)

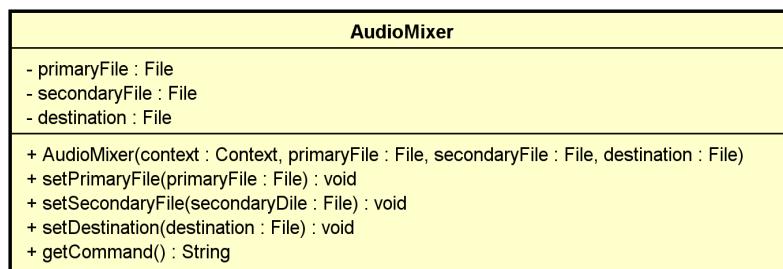


Figura 114: Classe Sivodim::Drama::Model::Screenplay::AudioMixer

3.17.11.1 Descrizione

Classe concreta che estende FfmpegConnector e ne implementa il metodo astratto getCommand per gestire la sovrapposizione di tracce audio.

3.17.11.2 Utilizzo

Viene utilizzata dalla classe ScreenplayImpl nella fase di esportazione per sovrapporre la traccia audio contenente la sintesi delle battute alla soundtrack del capitolo.

3.17.11.3 Classi ereditate

- Sivodim::Drama::Model::Screenplay::FfmpegConnector.

3.17.11.4 Attributi

3.17.11.4.1 primaryFile

- **Accesso:** private;
- **Tipo:** File;
- **Descrizione:** File principale che determina la lunghezza del file audio generato.

3.17.11.4.2 secondaryFile

- **Accesso:** private;
- **Tipo:** File;
- **Descrizione:** File contenente la traccia audio che viene sovrapposta al primaryFile.

3.17.11.4.3 destination

- **Accesso:** private;
- **Tipo:** File;
- **Descrizione:** Riferimento a File che determina la posizione del salvataggio dell'audio finale.

3.17.11.5 Metodi

3.17.11.5.1 AudioMixer(Context context, File primaryFile, File secondaryFile, File destination)

- **Accesso:** public;
- **Argomenti:**
 - context: il contesto della view;
 - primaryFile: il file principale che determina la durata dell'audio finale;
 - secondaryFile: il file secondario che deve essere sovrapposto;
 - destination: il file di destinazione.
- **Descrizione:** Costruttore che memorizza il file principale, il file secondario e il file di destinazione.

3.17.11.5.2 setPrimaryFile(File primaryFile)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - primaryFile: il file principale che determina la durata dell'audio finale.
- **Descrizione:** Metodo setter che riassegna il file principale.

3.17.11.5.3 setSecondaryFile(File secondaryFile)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - secondaryFile: il file secondario che deve sovrapporsi al principale.
- **Descrizione:** Metodo setter che riassegna il file secondario, che dovrà sovrapporsi al principale.

3.17.11.5.4 setDestination(File destination)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - destination: il file di destinazione in cui verrà salvato l'audio finale.
- **Descrizione:** Metodo setter che riassegna il file di destinazione in cui viene salvato l'audio finale.

3.17.11.5.5 getCommand()

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Descrizione:** Definizione del metodo astratto ereditato da FfmpegConnector che restituisce la String con il comando ffmpeg per la sovrapposizione di file audio.

3.17.12 VideoConcatenator (class)

VideoConcatenator
<ul style="list-style-type: none"> - destination : File - files : ArrayList<File> - listTxt : File
<ul style="list-style-type: none"> + VideoConcatenator(context : Context, destination : File) + add(file : File) : void + createFileList() : void + getCommand() : String

Figura 115: Classe Sivodim::Drama::Model::Screenplay::VideoConcatenator

3.17.12.1 Descrizione

Classe concreta che estende FfmpegConnector e ne implementa il metodo astratto getCommand per gestire la concatenazione di file video di tipo MP4_G.

3.17.12.2 Utilizzo

Viene utilizzata dalla classe ScreenplayImpl nella fase di esportazione per concatenare i file video generati a partire dalle battute e dalle immagini e i suoni associati.

3.17.12.3 Classi ereditate

- Sivodim::Drama::Model::Screenplay::FfmpegConnector.

3.17.12.4 Attributi

3.17.12.4.1 files

- **Accesso:** private;
- **Tipo:** Vector<File>;
- **Descrizione:** Vector di File contenente la lista dei file video da concatenare.

3.17.12.4.2 destination

- **Accesso:** public;
- **Tipo:** File;
- **Descrizione:** File di destinazione in cui salvare il file video finale.

3.17.12.4.3 listTxt

- **Accesso:** private;
- **Tipo:** File;
- **Descrizione:** File di utilità, necessario a FFmpeg per effettuare l'operazione di concatenazione.

3.17.12.5 Metodi

3.17.12.5.1 VideoConcatenator(Context context, File destination)

- **Accesso:** public;
- **Argomenti:**
 - context: il contesto della view.
 - destination: file di destinazione del processo di creazione del file video.
- **Descrizione:** Costruttore che prende il contesto e il percorso di destinazione della concatenazione, inizializzando a null la lista di file da concatenare.

3.17.12.5.2 add(File file)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - file: il File che deve essere aggiunto alla coda di concatenazione.
- **Descrizione:** Aggiunge un file alla coda di concatenazione.

3.17.12.5.3 `createFileList()`

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Descrizione:** Crea un file testuale contenente l'elenco dei percorsi per i file video che devono essere concatenati, nel rispetto della sintassi usata da FFmpeg.

3.17.12.5.4 `getCommand()`

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Definizione del metodo astratto ereditato da FfmpegConnector. Restituisce una String contenente il comando per FFmpeg per la concatenazione dei file video presenti nello stato.

3.17.13 VideoOverlayer (class)

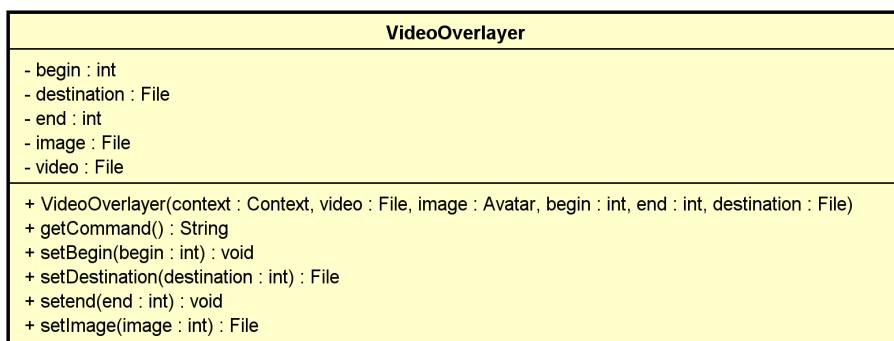


Figura 116: Classe Sivodim::Drama::Model::Screenplay::VideoOverlayer

3.17.13.1 Descrizione

Classe di utilità che estende FfmpegConnector necessaria per sovrapporre, nel contesto di un video, le immagini avatar_G relative ai personaggi allo sfondo associato a un capitolo .

3.17.13.2 Utilizzo

Viene utilizzata dalla classe FfmpegConnector per eseguire correttamente l'esportazione in video dello sceneggiato.

3.17.13.3 Classi ereditate

- Sivodim::Drama::Model::Screenplay::FfmpegConnector.

3.17.13.4 Attributi

3.17.13.4.1 video

- **Accesso:** private;
- **Tipo:** File;
- **Descrizione:** Tiene traccia del File video a cui sovrapporre l'immagine_G.

3.17.13.4.2 destination

- **Accesso:** private;
- **Tipo:** File;
- **Descrizione:** Il File di destinazione dove viene salvato il video convertito in MP4_G.

3.17.13.4.3 image

- **Accesso:** private;
- **Tipo:** File;
- **Descrizione:** Il File immagine che deve essere sovrapposto al video di origine_G.

3.17.13.4.4 begin

- **Accesso:** private;
- **Tipo:** int;
- **Descrizione:** intero che rappresenta il punto di inizio del video_G.

3.17.13.4.5 end

- **Accesso:** private;
- **Tipo:** int;
- **Descrizione:** intero che rappresenta il punto di fine del video_G.

3.17.13.5 Metodi

3.17.13.5.1 VideoOverlayer(Context context, File video, Avatar image, int begin, int end, File destination)

- **Accesso:** public;
- **Argomenti:**
 - context: il contesto della view;
 - video: il file video di origine_G;
 - image: il file immagine da sovrapporre al video;
 - begin: punto di inizio del video;
 - end: punto di fine del video;
 - destination: il file di destinazione dove viene salvato il video.
- **Descrizione:** Costruttore che riceve in input il File da convertire in MP3 e la sua destinazione.

3.17.13.5.2 setImage(File image)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - image: il file immagine da sovrapporre.
- **Descrizione:** Metodo setter che reimposta il file immagine da sovrapporre.

3.17.13.5.3 setBegin(int begin)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - begin: l'intero che rappresenta l'inizio del video.
- **Descrizione:** Metodo setter che reimposta l'attributo begin.

3.17.13.5.4 setEnd(int end)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - end: l'intero che rappresenta la fine del video.
- **Descrizione:** Metodo setter che reimposta l'attributo end.

3.17.13.5.5 setDestination(File destination)

- **Accesso:** public;
- **Tipo di ritorno:** void;
- **Argomenti:**
 - destination: il File in cui verrà salvata la conversione.
- **Descrizione:** Metodo setter che reimposta il File di destinazione.

3.17.13.5.6 getCommand()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Definizione del metodo astratto ereditato da FfmpegConnector, che restituisce il comando da dare a ffmpeg per convertire il file di cui si ha un riferimento in MP4.

3.17.14 ImageVideoConverter (class)

ImageVideoConverter
<ul style="list-style-type: none"> - destination : File - duration : int - image : File
<ul style="list-style-type: none"> + ImageVideoConverter(context : Context, background : Background, duration : int, destination : File) + getCommand() : void

Figura 117: Classe Sivodim::Drama::Model::Screenplay::ImageVideoConverter

3.17.14.1 Descrizione

Classe di utilità che estende FfmpegConnector al fine di creare un *loop* video di una certa durata a partire da un'immagine.

3.17.14.2 Utilizzo

Viene utilizzata dalla classe FFmpegConnector creare un video a partire da un file immagine.

3.17.14.3 Classi ereditate

- Sivodim::Drama::Model::Screenplay::FfmpegConnector.

3.17.14.4 Attributi

3.17.14.4.1 duration

- **Accesso:** private;
- **Tipo:** int;
- **Descrizione:** Intero che definisce la durata del file video.

3.17.14.4.2 destination

- **Accesso:** private;
- **Tipo:** File;
- **Descrizione:** Il File di destinazione dove viene salvato il video convertito in MP4_G.

3.17.14.4.3 image

- **Accesso:** private;
- **Tipo:** File;
- **Descrizione:** Il File immagine da cui viene originato il file video.

3.17.14.5 Metodi

3.17.14.5.1 ImageVideoConverter(Context context, Background background, int duration, File destination)

- **Accesso:** public;
- **Argomenti:**
 - context: il contesto della view;
 - background: il file immagine da cui originare il video;
 - duration: durata del file video da creare;
 - destination: il file di destinazione dove viene salvato il video.
- **Descrizione:** Costruttore che riceve in input il immagine da cui originare il video la destinazione del video.

3.17.14.5.2 getCommand()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Definizione del metodo astratto ereditato da FfmpegConnector, che restituisce il comando da dare a ffmpeg per convertire il file di cui si ha un riferimento in MP4.

3.17.15 AudioVideoMixer (class)



Figura 118: Classe Sivodim::Drama::Model::Screenplay::AudioVideoMixer

3.17.15.1 Descrizione

Classe di utilità che estende FfmpegConnector al fine di aggiungere a un file video una traccia audio.

3.17.15.2 Utilizzo

Viene utilizzata dalla classe FFmpegConnector per generare un file video completo di audio.

3.17.15.3 Classi ereditate

- Sivodim::Drama::Model::Screenplay::FfmpegConnector.

3.17.15.4 Attributi

3.17.15.4.1 audio

- **Accesso:** private;
- **Tipo:** File;
- **Descrizione:** Traccia audio da aggiungere al file video.

3.17.15.4.2 destination

- **Accesso:** private;
- **Tipo:** File;
- **Descrizione:** Il File di destinazione dove viene salvato il video convertito in MP4_G.

3.17.15.4.3 video

- **Accesso:** private;
- **Tipo:** File;
- **Descrizione:** Il File video a cui deve essere aggiunta la traccia audio.

3.17.15.5 Metodi

3.17.15.5.1 AudioVideoMixer(Context context, File video, File audio, File destination)

- **Accesso:** public;
- **Argomenti:**
 - context: il contesto della view;
 - video: file di tipo video;
 - audio: file di tipo audio;
 - destination: il file di destinazione dove viene salvato il video.
- **Descrizione:** Costruttore che riceve in input il immagine da cui originare il video la destinazione del video.

3.17.15.5.2 getCommand()

- **Accesso:** public;
- **Tipo di ritorno:** String;
- **Descrizione:** Definizione del metodo astratto ereditato da FfmpegConnector, che restituisce il comando da dare a ffmpeg per convertire il file di cui si ha un riferimento in MP4.

4 Diagrammi di sequenza

4.1 Creazione sceneggiato

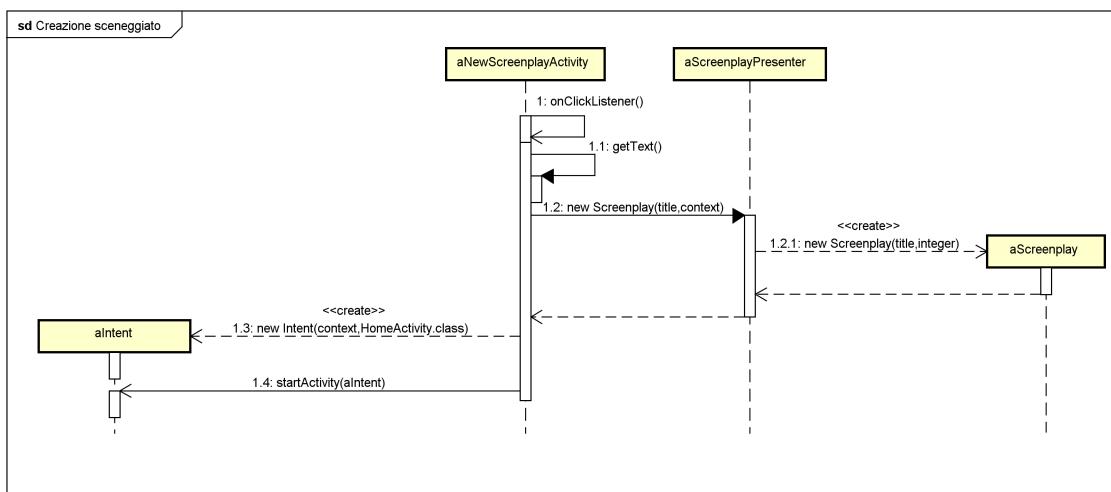


Figura 119: Diagramma di sequenza per la creazione di uno sceneggiato

Il diagramma mostra le interazioni fra gli oggetti del pacchetto Sivodim::Drama al momento della creazione di un nuovo sceneggiato.

- **aNewScreenplayActivity:** rappresenta un’istanza dell’activity NewScreenplayActivity, che si occupa di gestire l’interazione con l’utente per la creazione di nuovi sceneggiati. Alla pressione del pulsante di conferma da parte dell’utente, viene invocato l’onClickListner che invia al aScreenplayPresenter la richiesta di costruire lo sceneggiato a partire dai dati trovati. Una volta creato lo sceneggiato, viene creata una Intent per cambiare activity e, attraverso questa, viene mostrata l’activity contenente la lista degli sceneggiati creati;
- **aScreenplayPresenter:** rappresenta l’istanza di un oggetto di tipo ScreenplayPresenter che, ricevuta la richiesta di aggiunta di un nuovo sceneggiato, si occupa di istanziarlo.
- **aScreenplay:** rappresenta l’istanza di un oggetto di tipo Screenplay;
- **aIntent:** rappresenta l’istanza di un oggetto di tipo Intent, che viene utilizzato in questo contesto per passare ad un’altra activity.

4.2 Creazione capitolo

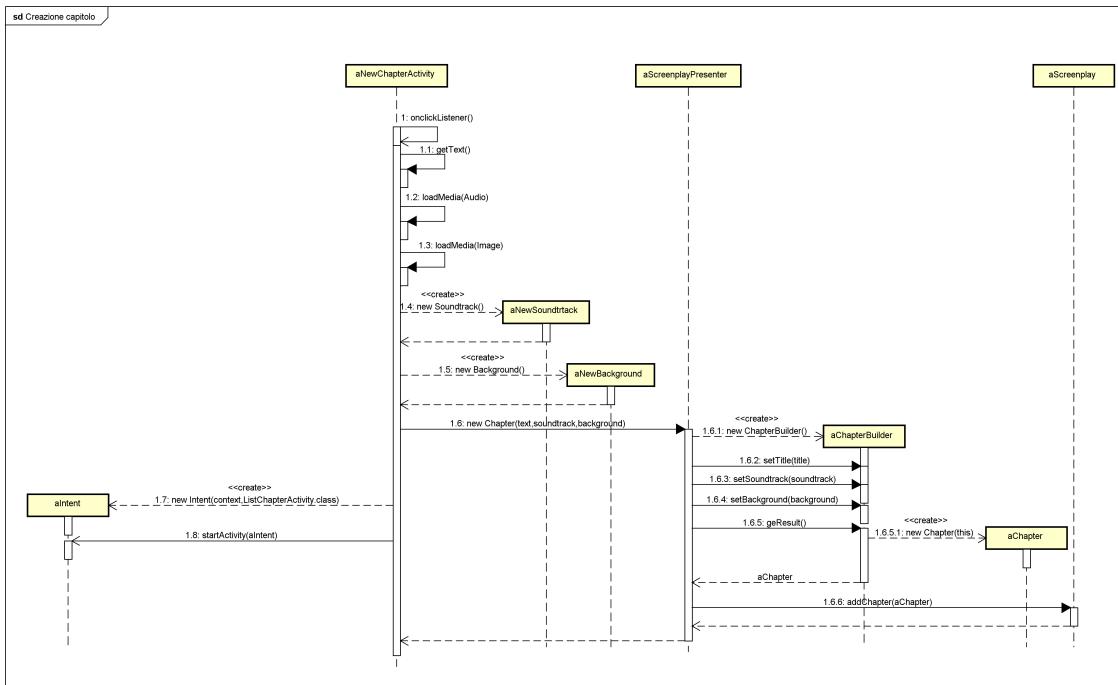


Figura 120: Diagramma di sequenza per la creazione di un capitolo

Il diagramma mostra le interazioni fra gli oggetti del pacchetto Sivodim::Drama al momento della creazione di un capitolo.

- **aNewChapterActivity:** rappresenta un’istanza dell’activity NewChapterActivity, che si occupa di gestire l’interazione con l’utente per la creazione un nuovo capitolo. Alla pressione del pulsante di conferma da parte dell’utente, viene invocato l’onClickListner che invia al aScreenplayPresenter la richiesta di costruisce il capitolo a partire dai dati trovati. Una volta creato il capitolo, viene creata una Intent per cambiare activity e, attraverso questa, viene mostrata l’activity contenente la lista dei capitoli presenti nello sceneggiato;
- **aScreenplayPresenter:** rappresenta l’istanza di un oggetto di tipo ScreenplayPresenter che, ricevuta la richiesta di aggiunta di un nuovo capitolo, si occupa di istanziarlo ed aggiungerlo in coda alla lista dei capitoli dello sceneggiato di cui ha un riferimento. Per istanziare il capitolo crea anzitutto un ChapterBuilder, dove imposta i parametri passati dall’activity;
- **aChapterBuilder:** rappresenta l’istanza di un oggetto di tipo ChapterBuilder, che si occupa di costruire un Chapter;
- **aScreenplay:** rappresenta l’istanza di un oggetto di tipo screenplay che contiene una lista di capitoli e prevede un metodo per poterne aggiungere in coda;
- **aChapter:** rappresenta l’istanza di un oggetto di tipo Chapter, caratterizzato da un titolo, una soundtrack e un background;
- **aSoundtrack:** rappresenta l’istanza di un oggetto di tipo Soundtrack associata ad un capitolo;

- **aBackground:** rappresenta l'istanza di un oggetto di tipo Background associata ad un capitolo;
- **aIntent:** rappresenta l'istanza di un oggetto di tipo Intent, che viene utilizzato in questo contesto per passare ad un'altra activity.

4.3 Creazione personaggio

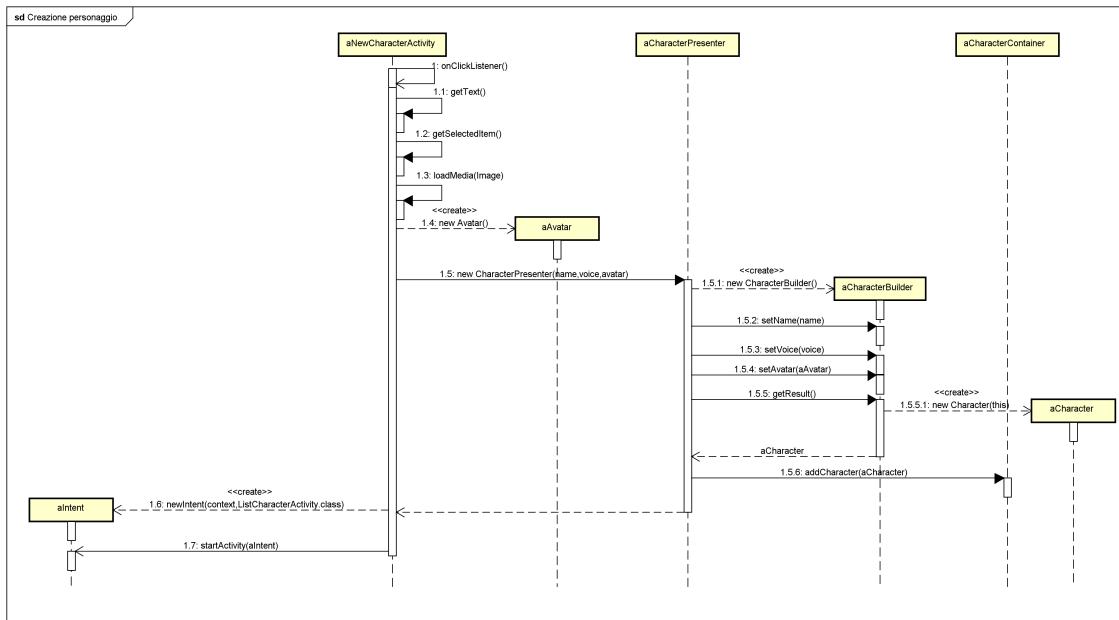


Figura 121: Diagramma di sequenza per la creazione di un personaggio

Il diagramma mostra le interazioni fra gli oggetti del pacchetto Sivodim::Drama al momento della creazione di un nuovo personaggio.

- **aNewCharacterActivity**: rappresenta un’istanza dell’activity NewCharacterActivity, che si occupa di gestire l’interazione con l’utente per la creazione di nuovi personaggi. Alla pressione del pulsante di conferma da parte dell’utente, viene invocato l’onClickListener che invia al aCharacterPresenter la richiesta di costruisce il personaggio a partire dai dati trovati. Una volta creato il personaggio, viene creata una Intent per cambiare activity e, attraverso questa, viene mostrata l’activity contenente la lista dei personaggi presenti nello sceneggiato;
- **aCharacterPresenter**: rappresenta l’istanza di un oggetto di tipo CharacterPresenter che, ricevuta la richiesta di aggiunta di un nuovo personaggio, si occupa di istanziarlo e di aggiungerlo allo sceneggiato di cui ha un riferimento. Per istanziare il personaggio crea anzitutto un CharacterBuilder, dove imposta i parametri passati dall’activity;
- **aCharacterBuilder**: rappresenta l’istanza di un oggetto di tipo CharacterBuilder, che si occupa di costruire un Character;
- **aCharacterContainer**: rappresenta l’istanza di un oggetto di tipo CharacterContainer, che gestisce i personaggi dello sceneggiato e permette di ottenere un loro riferimento a partire dal nome che li identifica;
- **aCharacter**: rappresenta l’istanza di un oggetto di tipo personaggio, caratterizzata da un nome, una voce e un avatar;
- **aAvatar**: rappresenta l’istanza di un oggetto di tipo Avatar associato al personaggio;
- **alIntent**: rappresenta l’istanza di un oggetto di tipo Intent, che viene utilizzato in questo contesto per passare ad un’altra activity.

4.4 Creazione battuta

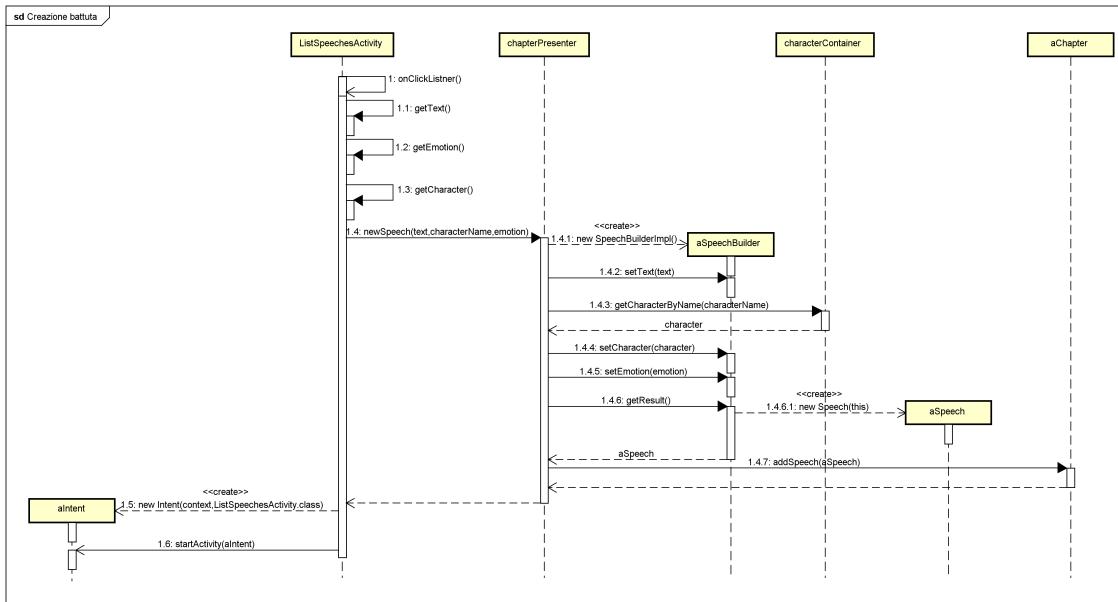


Figura 122: Diagramma di sequenza per la creazione di una battuta

Il diagramma mostra le interazioni fra gli oggetti del pacchetto Sivodim::Drama al momento della creazione di una nuova battuta.

- **aNewSpeechActivity:** rappresenta un’istanza dell’activity NewSpeechActivity, che si occupa di gestire l’interazione con l’utente per la creazione di nuove battute. Alla pressione del pulsante di conferma da parte dell’utente, viene invocato l’onClickListner che invia al chapterPresenter la richiesta di costruire la battuta a partire dai dati trovati. Una volta creata la battuta ed inserita nel capitolo, viene creata una Intent per cambiare activity e, attraverso questa, viene mostrata l’activity contenente la lista delle battute presenti nel capitolo;
- **chapterPresenter:** rappresenta l’istanza di un oggetto di tipo ChapterPresenter che, ricevuta la richiesta di aggiunta di una nuova battuta, si occupa di istanziarla ed aggiungerla in coda al capitolo di cui ha un riferimento. Per istanziare la battuta crea anzitutto uno SpeechBuilder, dove imposta i parametri passati dall’activity;
- **aSpeechBuilder:** rappresenta l’istanza di un oggetto di tipo SpeechBuilder, che si occupa di costruire una Speech;
- **characterContainer:** rappresenta l’istanza di un oggetto di tipo characterContainer, che gestisce i personaggi dello sceneggiato e permette di ottenere un loro riferimento a partire dal nome che li identifica;
- **aSpeech:** rappresenta l’istanza di un oggetto di tipo battuta, caratterizzata da un testo, un personaggio e un’emozione associata;
- **aChapter:** rappresenta l’istanza di un oggetto di tipo Chapter, che contiene una lista di battute e prevede un metodo per poterne aggiungere in coda;
- **aIntent:** rappresenta l’istanza di un oggetto di tipo Intent, che viene utilizzato in questo contesto per passare ad un’altra activity.

4.5 Visualizzazione battute

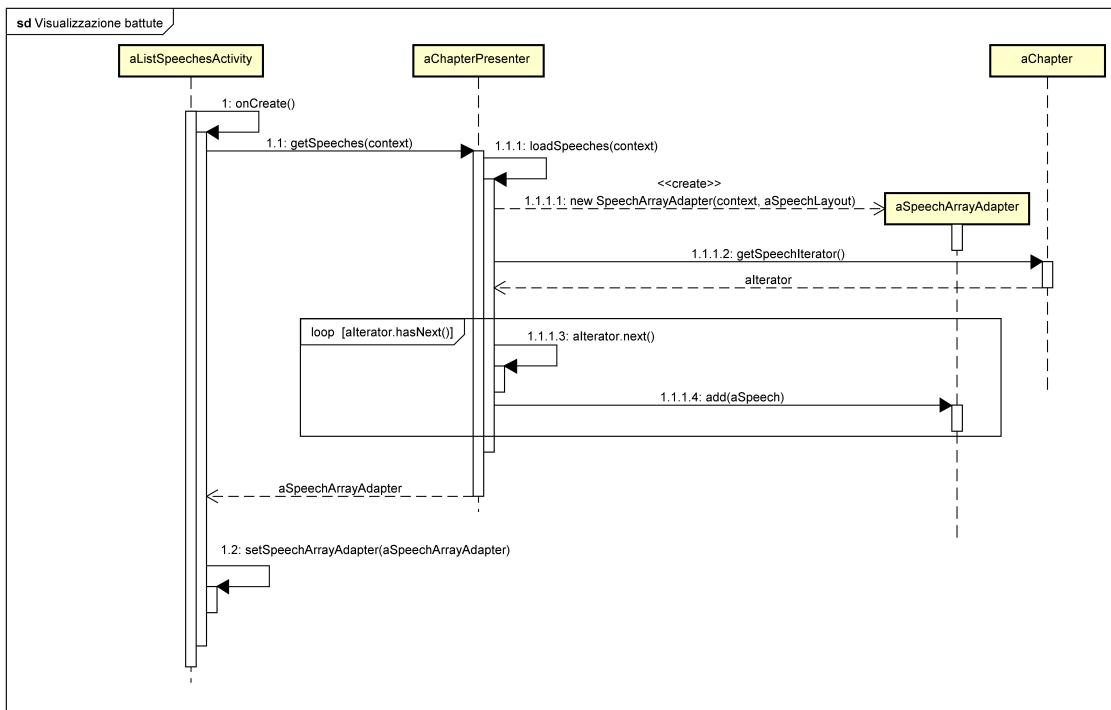


Figura 123: Diagramma di sequenza per la visualizzazione di un insieme di battute

Il diagramma mostra le interazioni fra gli oggetti del pacchetto Sivodim::Drama al momento della visualizzazione di un insieme di battute.

- **aListSpeechesActivity**: rappresenta un’istanza dell’activity ListSpeechesActivity, che si occupa di visualizzare a schermo l’insieme delle battute di un capitolo selezionato;
- **aChapterPresenter**: rappresenta l’istanza di un oggetto di tipo ChapterPresenter che, ricevuta la richiesta di visualizzazione delle battute di un capitolo, si occupa di creare un ArrayAdapter dove caricare le battute e renderle visualizzabili a video;
- **aChapter**: rappresenta l’istanza di un oggetto di tipo Chapter, che si occupa di ritornare un Iteratore all’ArrayAdapter per poterlo scorrere e costruire;
- **aSpeechArrayAdapter**: rappresenta l’istanza di un oggetto di tipo ArrayAdapter che viene riempito per poter visualizzare a video le battute del capitolo selezionato;
- **alintent**: rappresenta l’istanza di un oggetto di tipo Intent, che viene utilizzato in questo contesto per passare ad un’altra activity.

4.6 Modifica battuta

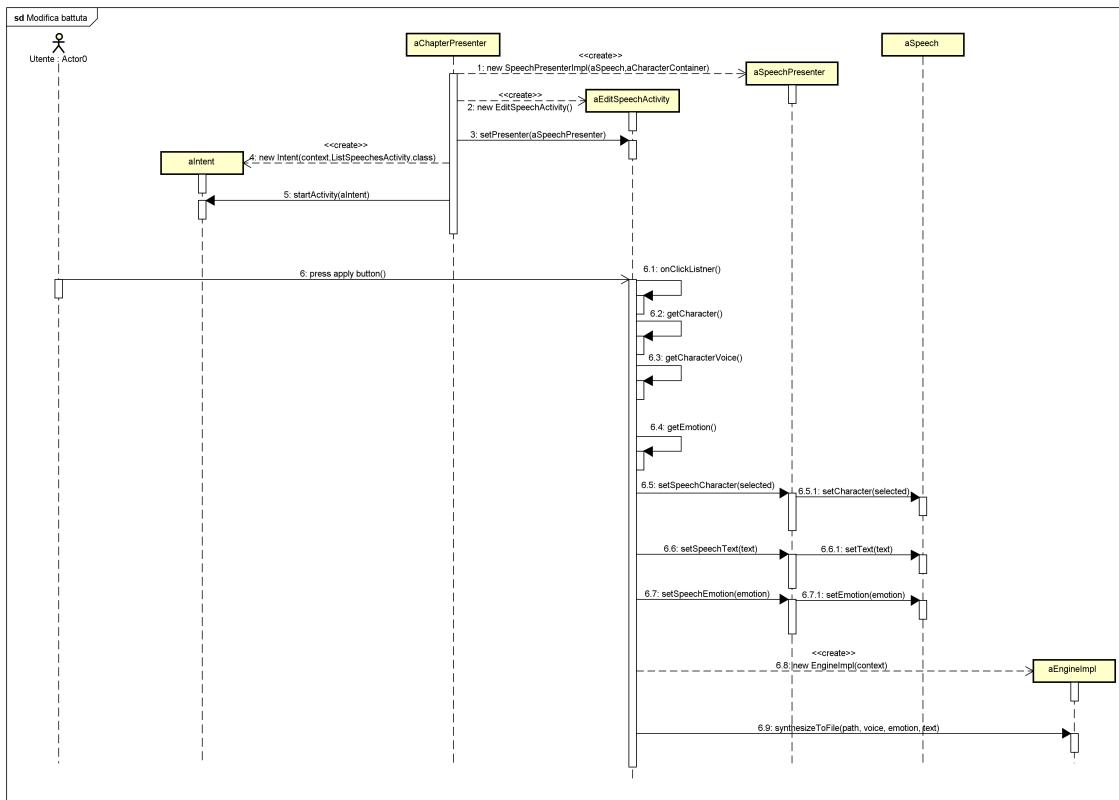


Figura 124: Diagramma di sequenza per la modifica di una battuta

Il diagramma mostra le interazioni fra gli oggetti del pacchetto Sivodim::Drama al momento della modifica di una battuta.

- **aEditSpeechActivity:** rappresenta un’istanza dell’activity EditSpeechActivity, che si occupa di gestire l’interazione con l’utente per la modifica di nuove battute. Alla pressione del pulsante di conferma da parte dell’utente, viene invocato l’onClickListner che invia al aChapterPresenter la richiesta di modificare la battuta. Una volta modificata la battuta ed inserita nel capitolo, viene creata una Intent per cambiare activity e, attraverso questa, viene mostrata l’activity contenente la lista delle battute presenti nel capitolo;
- **aChapterPresenter:** rappresenta l’istanza di un oggetto di tipo ChapterPresenter che, ricevuta la richiesta di modifica di una nuova battuta, si occupa di aggiungerla in coda al capitolo di cui ha un riferimento;
- **aSpeechPresenter:** rappresenta l’istanza di un oggetto di tipo SpeechPresenter, che si occupa di modificare una Speech a partire dai dati inseriti dall’utente;
- **aSpeech:** rappresenta l’istanza di un oggetto di tipo battuta, caratterizzata da un testo, un personaggio e un’emozione associata;
- **aIntent:** rappresenta l’istanza di un oggetto di tipo Intent, che viene utilizzato in questo contesto per passare ad un’altra activity.

4.7 Creazione voce

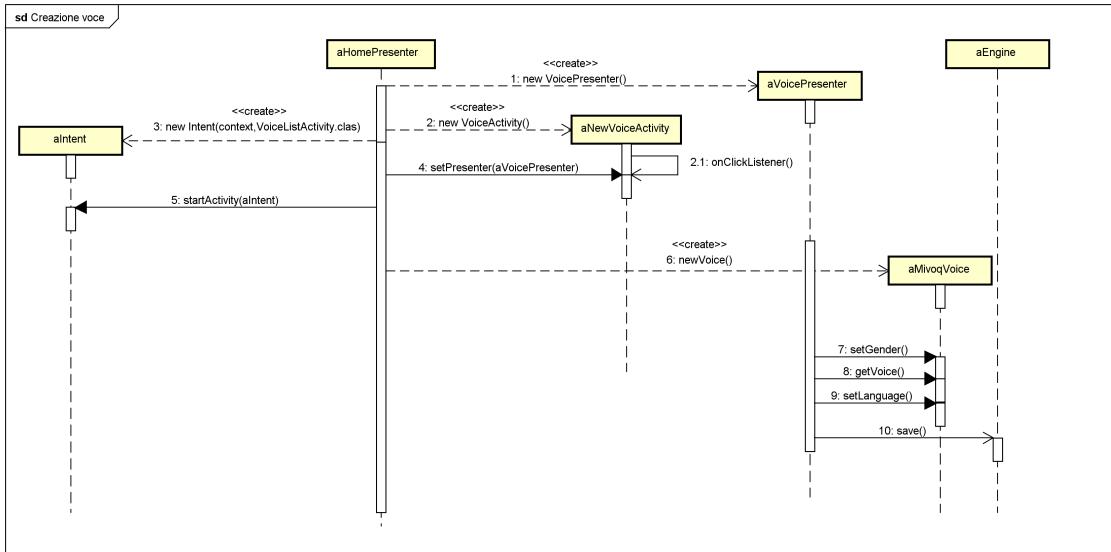


Figura 125: Diagramma di sequenza per la creazione di una voce

Il diagramma mostra le interazioni fra gli oggetti del pacchetto Sivodim::Libraries al momento della creazione di una nuova voce.

- **aHomePresenter**: rappresenta un’istanza di HomePresenter, che si occupa della creazione di una voce di default e di un VoicePresenter che passa alla NewVoiceActivity;
- **aNewVoiceActivity**: rappresenta un’istanza dell’activity NewVoiceActivity, che si occupa di gestire l’interazione con l’utente per la creazione di nuove voci. Alla pressione del pulsante di conferma da parte dell’utente, viene invocato l’onClickListner che invia al VoicePresenter la richiesta di settare i valori della voce creata di default da aHomePresenter a partire dai dati trovati. Una volta creata la voce e salvata, viene creata una Intent per cambiare activity e, attraverso questa, viene mostrata l’activity contenente la lista delle voci;
- **aVoicePresenter**: rappresenta l’istanza di un oggetto di tipo VoicePresenter che, ricevuta la richiesta di aggiunta di una nuova voce, si occupa di settarne i campi;
- **aMivoqVoice**: rappresenta l’istanza di un oggetto di tipo MivoqVoice creata da aHomePresenter e settata con i valori inseriti dall’utente;
- **aEngine**: rappresenta l’istanza di un oggetto di tipo Engine, invocata per salvare la voce appena creata;
- **aintent**: rappresenta l’istanza di un oggetto di tipo Intent, che viene utilizzato in questo contesto per passare ad un’altra activity.

4.8 Modifica voce

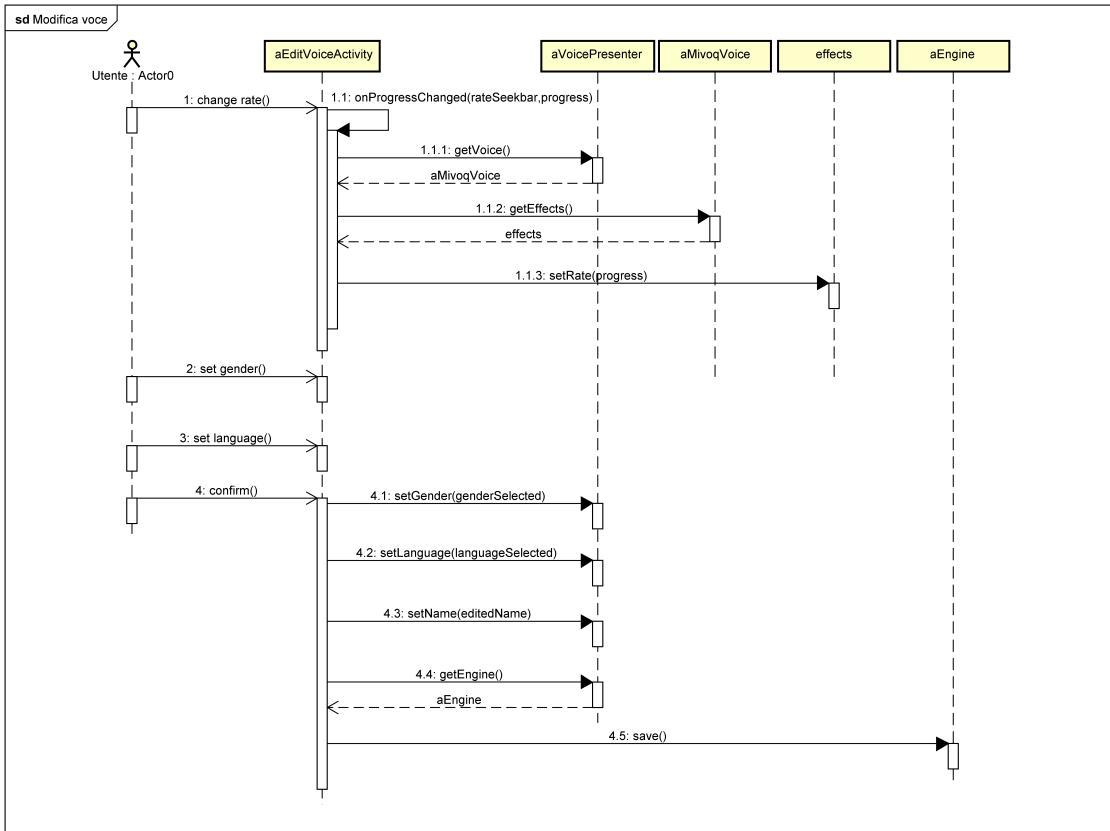


Figura 126: Diagramma di sequenza per la modifica di una voce

Il diagramma mostra le interazioni fra gli oggetti del pacchetto Sivodim::Libraries al momento della modifica di una nuova voce.

- **aEditVoiceActivity:** rappresenta un'istanza dell'activity EitVoiceActivity, che si occupa di gestire l'interazione con l'utente per la modifica di voci. Alla pressione del pulsante di conferma da parte dell'utente, viene invocato l'`onProgressChanged` che invia al VoicePresenter la richiesta di modificare i valori della voce creata;
- **aVoicePresenter:** rappresenta l'istanza di un oggetto di tipo VoicePresenter che, ricevuta la richiesta di modifica di una nuova voce, si occupa di settarne i campi;
- **aMivoqVoice:** rappresenta l'istanza di un oggetto di tipo MivoqVoice da modificare secondo i valori inseriti dall'utente;
- **aEngine:** rappresenta l'istanza di un oggetto di tipo Engine, invocata per salvare la voce appena creata;
- **effects:** rappresenta l'istanza di un oggetto di tipo Effect, che ritorna gli effetti associati alla voce richiesta.

4.9 Richiesta di sintesi vocale

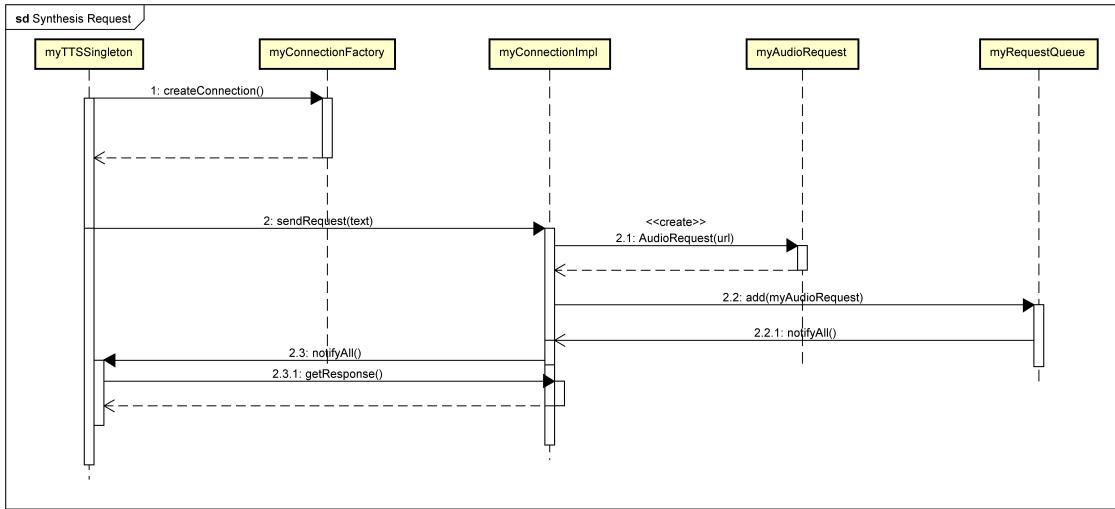


Figura 127: Diagramma di sequenza per la richiesta di sintesi vocale

Il diagramma descrive l'interazione tra gli oggetti quando è necessario inoltrare una richiesta di sintesi vocale di un testo al server di MIVOQ s.r.l..

- **myTTSSingleton**: rappresenta l'istanza di un oggetto di tipo MivoqTTSSingleton, che si occupa di ricevere una richiesta di sintesi vocale, e di richiedere l'istanziazione di oggetti di tipo MivoqConnection per l'invio di richieste ai server di MIVOQ s.r.l.;
- **myConnectionFactory**: rappresenta l'istanza di un oggetto di tipo MivoqConnectionFactory che ha il compito di istanziare gli oggetti di tipo MivoqConnectionImpl;
- **myConnectionImpl**: rappresenta l'istanza di un oggetto di tipo MivoqConnectionImpl, che si occupa di ricevere i parametri della sintesi vocale ed inoltrarli ai server di MIVOQ s.r.l. tramite la creazione di un oggetto di tipo AudioRequest. Inoltre si occupa di memorizzare e consegnare la risposta alla richiesta effettuata;
- **myAudioRequest**: rappresenta l'istanza di un oggetto di tipo AudioRequest, che si occupa dell'invio di una richiesta HTTP_G ad uno specifico url_G gestendo poi la ricezione del flusso di byte;
- **myRequestQueue**: rappresenta l'istanza di un oggetto di tipo RequestQueue, che si occupa della gestione della coda di richieste HTTP_G da inoltrare.

A Tracciamento

A.1 Tracciamento Requisiti - Classi

Requisito	Classi
-----------	--------

R0F1	Screenplay ScreenplayImpl ScreenplayPresenter ScreenplayPresenterImpl NewScreenplayInterface NewScreenplayActivity
R0F1.1	Screenplay ScreenplayImpl ScreenplayPresenter ScreenplayPresenterImpl NewScreenplayInterface NewScreenplayActivity
R1F8.9	Screenplay ScreenplayImpl ScreenplayPresenter ScreenplayPresenterImpl NewScreenplayInterface NewScreenplayActivity
R0F8.1	Character CharacterImpl CharacterBuilder CharacterBuilderImpl ChapterPresenter ChapterPresenterImpl NewCharacterInterface NewCharacterActivity
R0F2.3	MivoqTTSSingleton Engine
R0F5	MivoqTTSService
R0F7	MivoqTTSService
R0F2.2	EditVoiceActivity NewVoiceActivity
R0F8.1.1.1	NewCharacterInterface

R0F1.1.1	HomeInterface
R0F1.1.2	HomeInterface
R0F8.1.1	Character CharacterImpl CharacterBuilder CharacterBuilderImpl CharacterPresenter CharacterPresenterImpl NewCharacterInterface NewCharacterActivity EditCharacterInterface EditCharacterActivity
R0F8.7.4	EditCharacterInterface EditCharacterActivity
R1F8.7.1	Character CharacterImpl Image Avatar CharacterPresenter CharacterPresenterImpl EditCharacterInterface EditCharacterActivity
R0F8.7.7	Character CharacterImpl CharacterPresenter CharacterPresenterImpl EditCharacterInterface EditCharacterActivity
R0F8.7.7.1	NewCharacterInterface NewCharacterActivity EditCharacterInterface EditCharacterActivity
R1F8.7.3	Image Avatar

R1F8.7.2	Image Avatar
R0F8.7.8	Character CharacterImpl CharacterPresenter CharacterPresenterImpl EditCharacterInterface EditCharacterActivity
R0F2	VoicePresenter VoicePresenterImpl NewVoiceActivityInterface NewVoiceActivity MivoqVoice
R1F8.7.6	EditCharacterInterface EditCharacterActivity
R0F8.1.1.2	MivoqTTSSingleton
R0F8.10	Screenplay ScreenplayImpl Chapter ChapterImpl ChapterBuilder ChapterBuilderImpl ChapterPresenter ChapterPresenterImpl NewChapterInterface NewChapterActivity
R0F8	Screenplay ScreenplayImpl ScreenplayPresenter ScreenplayPresenterImpl ListChapterInterface ListChapterActivity EditChapterInterface EditChapterActivity

R0F8.11	ChapterPresenter ChapterPresenterImpl ListSpeechesInterface EditSpeechInterface EditSpeechActivity ListSpeechesActivity
R0F8.7	EditCharacterInterface EditCharacterActivity
R1F8.8	Character CharacterImpl CharacterPresenter CharacterPresenterImpl ListCharacterInterface ListCharacterActivity EditCharacterInterface EditCharacterActivity CharacterContainer CharacterAdapter
R0F8.6	ExportAlgorithm AudioExport VideoExport ScreenplayPresenter ScreenplayPresenterImpl ListChapterInterface ListChapterActivity
R0F8.3	Screenplay ScreenplayImpl ScreenplayPresenter ScreenplayPresenterImpl ListChapterInterface ListChapterActivity XMLParser
R0F8.2	Screenplay ScreenplayImpl Chapter ChapterImpl ListChapterInterface ListChapterActivity

R0F2.4	VoicePresenter VoicePresenterImpl NewVoiceActivityInterface NewVoiceActivity MivoqVoice Engine Enginelmpl
R0F2.1	VoicePresenter VoicePresenterImpl NewVoiceActivityInterface NewVoiceActivity MivoqVoice
R0F2.5	VoicePresenter VoicePresenterImpl NewVoiceActivityInterface NewVoiceActivity MivoqVoice
R0F9	VoicePresenter VoicePresenterImpl EditVoiceActivityInterface EditVoiceActivity MivoqVoice
R0F9.5	VoicePresenter VoicePresenterImpl EditVoiceActivityInterface EditVoiceActivity MivoqVoice Effect EffectImpl
R0F9.3	VoicePresenter VoicePresenterImpl EditVoiceActivityInterface EditVoiceActivity MivoqVoice
R0F9.4	MivoqVoice

R0F9.1	MivoqVoice Engine Enginelmpl
R0F9.2	Engine Enginelmpl
R0F8.10.1	NewChapterInterface NewChapterActivity
R0F8.10.1.1	EditChapterInterface EditChapterActivity
R0F8.10.1.2	EditChapterInterface EditChapterActivity
R1F8.11.11	EditChapterInterface EditChapterActivity
R0F8.11.3	ChapterPresenter ChapterPresenterImpl NewChapterInterface NewChapterActivity EditChapterInterface EditChapterActivity
R0F8.11.4	ChapterPresenter ChapterPresenterImpl SpeechPresenter SpeechPresenterImpl NewSpeechInterface NewSpeechActivity EditSpeechInterface EditSpeechActivity

R0F8.11.5	ChapterPresenter ChapterPresenterImpl SpeechPresenter SpeechPresenterImpl NewSpeechInterface NewSpeechActivity EditSpeechInterface EditSpeechActivity
R0F8.11.2	SpeechSound SpeechPresenter SpeechPresenterImpl
R0F8.11.6	Chapter ChapterImpl SpeechPresenter SpeechPresenterImpl ListSpeechesInterface ListSpeechesActivity
R0F8.11.9	Chapter ChapterImpl ChapterPresenter ChapterPresenterImpl ListSpeechesActivity
R1F8.11.3.2	ChapterPresenter ChapterPresenterImpl NewChapterInterface NewChapterActivity EditChapterInterface EditChapterActivity
R1F8.11.3.1	ChapterPresenter ChapterPresenterImpl NewChapterInterface NewChapterActivity EditChapterInterface EditChapterActivity
R1F8.11.3.1.2	NewChapterInterface NewChapterActivity EditChapterInterface

R1F8.11.3.1.1	ChapterPresenter ChapterPresenterImpl NewChapterInterface NewChapterActivity EditChapterInterface EditChapterActivity
R1F8.11.3.3	ChapterPresenter ChapterPresenterImpl NewChapterInterface NewChapterActivity EditChapterInterface EditChapterActivity
R1F8.11.3.3.2	NewChapterInterface NewChapterActivity EditChapterInterface EditChapterActivity
R1F8.11.3.3.1	ChapterPresenter ChapterPresenterImpl NewChapterInterface NewChapterActivity EditChapterInterface EditChapterActivity
R0F8.11.4.1	Speech SpeechImpl SpeechPresenter SpeechPresenterImpl SpeechBuilder SpeechBuilderImpl
R0F8.11.4.2	Speech SpeechImpl SpeechPresenter SpeechPresenterImpl NewSpeechInterface NewSpeechActivity EditSpeechInterface EditSpeechActivity SpeechBuilder SpeechBuilderImpl

R0F8.11.5.3	Speech SpeechImpl SpeechPresenter SpeechPresenterImpl EditSpeechInterface EditSpeechActivity
R0F8.11.5.1	EditSpeechInterface EditSpeechActivity
R0F8.11.5.2	Speech SpeechImpl SpeechPresenter SpeechPresenterImpl Emotion
R0F8.11.1	SpeechSound Speech SpeechImpl SpeechPresenter SpeechPresenterImpl EditSpeechInterface EditSpeechActivity
R0F8.11.10	SpeechSound Speech SpeechPresenter SpeechPresenterImpl EditSpeechInterface EditSpeechActivity
R0F8.6.1	ExportAlgorithm AudioExport
R2F8.6.3	ExportAlgorithm VideoExport
R0F8.6.1.1	ExportAlgorithm AudioExport
R0F8.6.1.2	ExportAlgorithm AudioExport

R0F10	ScreenplayImpl ScreenplayPresenterImpl NewScreenplayActivity XMLParser
R0F9.5.1	VoicePresenter VoicePresenterImpl NewVoiceActivityInterface NewVoiceActivity EditVoiceActivityInterface EditVoiceActivity Effect EffectImpl
R0F9.5.2	EditVoiceActivityInterface EditVoiceActivity MivoqVoice Effect EffectImpl VoiceListPresenter VoiceListPresenterImpl
R0F9.1.1	VoicePresenter VoicePresenterImpl NewVoiceActivityInterface NewVoiceActivity MivoqVoice Engine EngineImpl
R0F5.2	VoiceListActivityInterface VoiceListActivity Engine EngineImpl VoiceListPresenter VoiceListPresenterImpl
R0F2.1.5	VoicePresenter VoicePresenterImpl NewVoiceActivityInterface NewVoiceActivity MivoqVoice

R0F2.1.3	VoicePresenter VoicePresenterImpl NewVoiceActivityInterface NewVoiceActivity MivoqVoice
R0F2.1.4	VoicePresenter VoicePresenterImpl NewVoiceActivityInterface NewVoiceActivity MivoqVoice
R0F2.1.2	VoicePresenter VoicePresenterImpl NewVoiceActivityInterface NewVoiceActivity MivoqVoice
R0F2.1.6	VoicePresenter VoicePresenterImpl NewVoiceActivityInterface NewVoiceActivity MivoqVoice
R0F2.5.4	VoicePresenter VoicePresenterImpl NewVoiceActivityInterface NewVoiceActivity MivoqVoice
R0F2.5.3	VoicePresenter VoicePresenterImpl NewVoiceActivityInterface NewVoiceActivity MivoqVoice
R0F2.5.5	VoicePresenter VoicePresenterImpl NewVoiceActivityInterface NewVoiceActivity MivoqVoice

R0F2.5.1	VoicePresenter VoicePresenterImpl NewVoiceActivityInterface NewVoiceActivity MivoqVoice
R0F9.5.4	VoicePresenter VoicePresenterImpl NewVoiceActivityInterface NewVoiceActivity MivoqVoice Effect EffectImpl
R0F9.5.5	VoicePresenter VoicePresenterImpl NewVoiceActivityInterface NewVoiceActivity MivoqVoice Effect EffectImpl
R0F9.5.6	VoicePresenter VoicePresenterImpl NewVoiceActivityInterface NewVoiceActivity MivoqVoice Effect EffectImpl
R0F9.5.7	VoicePresenter VoicePresenterImpl EditVoiceActivityInterface EditVoiceActivity MivoqVoice Effect EffectImpl
R0F8.11.5.4	NewSpeechInterface NewSpeechActivity EditSpeechInterface EditSpeechActivity

R0F8.11.5.2.6	SpeechPresenter SpeechPresenterImpl NewSpeechInterface NewSpeechActivity EditSpeechInterface EditSpeechActivity Emotion EffectImpl Engine Enginelmpl
R0F8.11.5.2.1	SpeechPresenter SpeechPresenterImpl NewSpeechInterface NewSpeechActivity EditSpeechInterface EditSpeechActivity Emotion Effect EffectImpl Engine Enginelmpl
R0F8.11.5.2.5	SpeechPresenter SpeechPresenterImpl NewSpeechInterface NewSpeechActivity EditSpeechInterface EditSpeechActivity Emotion Effect EffectImpl Engine Enginelmpl

R0F8.11.5.2.4	SpeechPresenter SpeechPresenterImpl NewSpeechInterface NewSpeechActivity EditSpeechInterface EditSpeechActivity Emotion Effect EffectImpl Engine EngineImpl
R0F8.11.5.2.3	SpeechPresenter SpeechPresenterImpl NewSpeechInterface NewSpeechActivity EditSpeechInterface EditSpeechActivity Emotion Engine EngineImpl
R0F8.11.5.2.2	SpeechPresenter SpeechPresenterImpl NewSpeechInterface NewSpeechActivity EditSpeechInterface EditSpeechActivity Emotion Effect EffectImpl Engine EngineImpl
R0F8.11.4.1.1	CharacterPresenter CharacterPresenterImpl ListCharacterInterface ListCharacterActivity CharacterContainer CharacterAdapter
R0F8.6.2	ExportAlgorithm AudioExport VideoExport

R0F12	Screenplay ScreenplayImpl ScreenplayPresenter ScreenplayPresenterImpl
R0F2.5.2	Engine EnginemImpl
R2F15	TextToSpeech Engine EnginemImpl
R0F18	MivoqTTSSingleton
R0F19	Screenplay ScreenplayImpl

Tabella 2: Tracciamento Requisiti - Classi

A.2 Tracciamento Classi - Requisiti

Classe	Requisiti
Screenplay	R0F1 R0F1.1 R1F8.9 R0F8.10 R0F8 R0F8.3 R0F8.2 R0F12 R0F19
ScreenplayImpl	R0F1 R0F1.1 R1F8.9 R0F8.10 R0F8 R0F8.3 R0F8.2 R0F10 R0F12 R0F19
ExportAlgorithm	R0F8.6 R0F8.6.1 R2F8.6.3 R0F8.6.1.1 R0F8.6.1.2 R0F8.6.2
AudioExport	R0F8.6 R0F8.6.1 R0F8.6.1.1 R0F8.6.1.2 R0F8.6.2
VideoExport	R0F8.6 R2F8.6.3 R0F8.6.2

Character	R0F8.1 R0F8.1.1 R1F8.7.1 R0F8.7.7 R0F8.7.8 R1F8.8
CharacterImpl	R0F8.1 R0F8.1.1 R1F8.7.1 R0F8.7.7 R0F8.7.8 R1F8.8
CharacterBuilder	R0F8.1 R0F8.1.1
CharacterBuilderImpl	R0F8.1 R0F8.1.1
SpeechSound	R0F8.11.2 R0F8.11.1 R0F8.11.10
Image	R1F8.7.1 R1F8.7.3 R1F8.7.2
Avatar	R1F8.7.1 R1F8.7.3 R1F8.7.2
Chapter	R0F8.10 R0F8.2 R0F8.11.6 R0F8.11.9
ChapterImpl	R0F8.10 R0F8.2 R0F8.11.6 R0F8.11.9

ChapterBuilder	R0F8.10
ChapterBuilderImpl	R0F8.10
Speech	R0F8.11.4.1 R0F8.11.4.2 R0F8.11.5.3 R0F8.11.5.2 R0F8.11.1 R0F8.11.10
SpeechImpl	R0F8.11.4.1 R0F8.11.4.2 R0F8.11.5.3 R0F8.11.5.2 R0F8.11.1
ScreenplayPresenter	R0F1 R0F1.1 R1F8.9 R0F8 R0F8.6 R1F8.12 R0F8.3 R0F12
ScreenplayPresenterImpl	R0F1 R0F1.1 R1F8.9 R0F8 R0F8.6 R1F8.12 R0F8.3 R0F10 R0F12

ChapterPresenter	R0F8.1 R0F8.10 R0F8.11 R0F8.11.3 R0F8.11.4 R0F8.11.5 R0F8.11.9 R1F8.11.3.2 R1F8.11.3.1 R1F8.11.3.1.1 R1F8.11.3.3 R1F8.11.3.3.1
ChapterPresenterImpl	R0F8.1 R0F8.10 R0F8.11 R0F8.11.3 R0F8.11.4 R0F8.11.5 R0F8.11.9 R1F8.11.3.2 R1F8.11.3.1 R1F8.11.3.1.1 R1F8.11.3.3 R1F8.11.3.3.1
SpeechPresenter	R0F8.11.4 R0F8.11.5 R0F8.11.2 R0F8.11.6 R0F8.11.4.1 R0F8.11.4.2 R0F8.11.5.3 R0F8.11.5.2 R0F8.11.1 R0F8.11.10 R0F8.11.5.2.6 R0F8.11.5.2.1 R0F8.11.5.2.5 R0F8.11.5.2.4 R0F8.11.5.2.3 R0F8.11.5.2.2

SpeechPresenterImpl	R0F8.11.4 R0F8.11.5 R0F8.11.2 R0F8.11.6 R0F8.11.4.1 R0F8.11.4.2 R0F8.11.5.3 R0F8.11.5.2 R0F8.11.1 R0F8.11.10 R0F8.11.5.2.6 R0F8.11.5.2.1 R0F8.11.5.2.5 R0F8.11.5.2.4 R0F8.11.5.2.3 R0F8.11.5.2.2
CharacterPresenter	R0F8.1.1 R1F8.7.1 R0F8.7.7 R0F8.7.8 R1F8.8 R0F8.11.4.1.1
CharacterPresenterImpl	R0F8.1.1 R1F8.7.1 R0F8.7.7 R0F8.7.8 R1F8.8 R0F8.11.4.1.1
ListChapterInterface	R0F8 R0F8.6 R1F8.12 R0F8.3 R0F8.2
ListChapterActivity	R0F8 R0F8.6 R1F8.12 R0F8.3 R0F8.2

NewScreenplayInterface	R0F1 R0F1.1 R1F8.9
NewScreenplayActivity	R0F1 R0F1.1 R1F8.9 R0F10
ListSpeechesInterface	R0F8.11 R0F8.11.6
VoicePresenter	R0F2 R0F2.4 R0F2.1 R0F2.5 R0F9 R0F9.5 R0F9.3 R0F9.5.1 R0F9.1.1 R0F2.1.5 R0F2.1.3 R0F2.1.4 R0F2.1.2 R0F2.1.6 R0F2.5.4 R0F2.5.3 R0F2.5.5 R0F2.5.1 R0F9.5.4 R0F9.5.5 R0F9.5.6 R0F9.5.7

VoicePresenterImpl	R0F2 R0F2.4 R0F2.1 R0F2.5 R0F9 R0F9.5 R0F9.3 R0F9.5.1 R0F9.1.1 R0F2.1.5 R0F2.1.3 R0F2.1.4 R0F2.1.2 R0F2.1.6 R0F2.5.4 R0F2.5.3 R0F2.5.5 R0F2.5.1 R0F9.5.4 R0F9.5.5 R0F9.5.6 R0F9.5.7
NewVoiceActivityInterface	R0F2 R0F2.4 R0F2.1 R0F2.5 R0F9.5.1 R0F9.1.1 R0F2.1.5 R0F2.1.3 R0F2.1.4 R0F2.1.2 R0F2.1.6 R0F2.5.4 R0F2.5.3 R0F2.5.5 R0F2.5.1 R0F9.5.4 R0F9.5.5 R0F9.5.6

NewVoiceActivity	R0F2 R0F2.2 R0F2.4 R0F2.1 R0F2.5 R0F9.5.1 R0F9.1.1 R0F2.1.5 R0F2.1.3 R0F2.1.4 R0F2.1.2 R0F2.1.6 R0F2.5.4 R0F2.5.3 R0F2.5.5 R0F2.5.1 R0F9.5.4 R0F9.5.5 R0F9.5.6
EditVoiceActivityInterface	R0F2.2 R0F9 R0F9.5 R0F9.3 R0F9.5.1 R0F9.5.2 R0F9.5.7
EditVoiceActivity	R0F9 R0F9.5 R0F9.3 R0F9.5.1 R0F9.5.2 R0F9.5.7
VoiceListActivityInterface	R0F5.2
VoiceListActivity	R0F5.2

NewChapterInterface	R0F8.10 R0F8.10.1 R0F8.11.3 R1F8.11.3.2 R1F8.11.3.1 R1F8.11.3.1.2 R1F8.11.3.1.1 R1F8.11.3.3 R1F8.11.3.3.2 R1F8.11.3.3.1
NewChapterActivity	R0F8.10 R0F8.10.1 R0F8.11.3 R1F8.11.3.2 R1F8.11.3.1 R1F8.11.3.1.2 R1F8.11.3.1.1 R1F8.11.3.3 R1F8.11.3.3.2 R1F8.11.3.3.1
EditChapterInterface	R0F8 R0F8.10.1.1 R0F8.10.1.2 R1F8.11.11 R0F8.11.3 R1F8.11.3.2 R1F8.11.3.1 R1F8.11.3.1.2 R1F8.11.3.1.1 R1F8.11.3 R1F8.11.3.3.2 R1F8.11.3.3.1

EditChapterActivity	R0F8 R0F8.10.1.1 R0F8.10.1.2 R1F8.11.11 R0F8.11.3 R1F8.11.3.2 R1F8.11.3.1 R1F8.11.3.1.1 R1F8.11.3.3 R1F8.11.3.3.2 R1F8.11.3.3.1
NewSpeechInterface	R0F8.11.4 R0F8.11.5 R0F8.11.4.2 R0F8.11.5.4 R0F8.11.5.2.6 R0F8.11.5.2.1 R0F8.11.5.2.5 R0F8.11.5.2.4 R0F8.11.5.2.3 R0F8.11.5.2.2
NewSpeechActivity	R0F8.11.4 R0F8.11.5 R0F8.11.4.2 R0F8.11.5.4 R0F8.11.5.2.6 R0F8.11.5.2.1 R0F8.11.5.2.5 R0F8.11.5.2.4 R0F8.11.5.2.3 R0F8.11.5.2.2

EditSpeechInterface	R0F8.11 R0F8.11.4 R0F8.11.5 R0F8.11.4.2 R0F8.11.5.3 R0F8.11.5.1 R0F8.11.1 R0F8.11.10 R0F8.11.5.4 R0F8.11.5.2.6 R0F8.11.5.2.1 R0F8.11.5.2.5 R0F8.11.5.2.4 R0F8.11.5.2.3 R0F8.11.5.2.2
EditSpeechActivity	R0F8.11 R0F8.11.4 R0F8.11.5 R0F8.11.4.2 R0F8.11.5.3 R0F8.11.5.1 R0F8.11.1 R0F8.11.10 R0F8.11.5.4 R0F8.11.5.2.6 R0F8.11.5.2.1 R0F8.11.5.2.5 R0F8.11.5.2.4 R0F8.11.5.2.3 R0F8.11.5.2.2
NewCharacterInterface	R0F8.1 R0F8.1.1 R0F8.7.7.1
NewCharacterActivity	R0F8.1.1.1 R0F8.1 R0F8.1.1 R0F8.7.7.1
ListCharacterInterface	R1F8.8 R0F8.11.4.1.1

ListCharacterActivity	R1F8.8 R0F8.11.4.1.1
EditCharacterInterface	R0F8.1.1 R0F8.7.4 R1F8.7.1 R0F8.7.7 R0F8.7.7.1 R0F8.7.8 R1F8.7.6 R0F8.7 R1F8.8
EditCharacterActivity	R0F8.1.1 R0F8.7.4 R1F8.7.1 R0F8.7.7 R0F8.7.7.1 R0F8.7.8 R1F8.7.6 R0F8.7 R1F8.8

MivoqVoice	R0F2 R0F2.4 R0F2.1 R0F2.5 R0F9 R0F9.5 R0F9.3 R0F9.4 R0F9.1 R0F9.5.2 R0F9.1.1 R0F2.1.5 R0F2.1.3 R0F2.1.4 R0F2.1.2 R0F2.1.6 R0F2.5.4 R0F2.5.3 R0F2.5.5 R0F2.5.1 R0F9.5.4 R0F9.5.5 R0F9.5.6 R0F9.5.7
Emotion	R0F8.11.5.2 R0F8.11.5.2.6 R0F8.11.5.2.1 R0F8.11.5.2.5 R0F8.11.5.2.4 R0F8.11.5.2.3 R0F8.11.5.2.2
Effect	R0F9.5 R0F9.5.1 R0F9.5.2 R0F9.5.4 R0F9.5.5 R0F9.5.6 R0F9.5.7 R0F8.11.5.2.1 R0F8.11.5.2.5 R0F8.11.5.2.4 R0F8.11.5.2.2

EffectImpl	R0F9.5 R0F9.5.1 R0F9.5.2 R0F9.5.4 R0F9.5.5 R0F9.5.6 R0F9.5.7 R0F8.11.5.2.6 R0F8.11.5.2.1 R0F8.11.5.2.5 R0F8.11.5.2.4 R0F8.11.5.2.2
TextToSpeech	R2F15
MivoqTTSSingleton	R0F2.3 R0F8.1.1.2 R0F18
Engine	R0F2.3 R0F2.4 R0F9.1 R0F9.2 R0F9.1.1 R0F5.2 R0F8.11.5.2.6 R0F8.11.5.2.1 R0F8.11.5.2.5 R0F8.11.5.2.4 R0F8.11.5.2.3 R0F8.11.5.2.2 R0F2.5.2 R2F15

Enginelmpl	R0F2.4 R0F9.1 R0F9.2 R0F9.1.1 R0F5.2 R0F8.11.5.2.6 R0F8.11.5.2.1 R0F8.11.5.2.5 R0F8.11.5.2.4 R0F8.11.5.2.3 R0F8.11.5.2.2 R0F2.5.2 R2F15
ListSpeechesActivity	R0F8.11 R0F8.11.6 R0F8.11.9
CharacterContainer	R1F8.8 R0F8.11.4.1.1
XMLParser	R0F8.3 R0F10
VoiceListPresenter	R0F9.5.2 R0F5.2
VoiceListPresenterImpl	R0F9.5.2 R0F5.2
SpeechBuilder	R0F8.11.4.1 R0F8.11.4.2
SpeechBuilderImpl	R0F8.11.4.1 R0F8.11.4.2
CharacterArrayAdapter	R1F8.8 R0F8.11.4.1.1
MivoqTTSService	R0F5 R0F7

HomeInterface	R0F1.1.1 R0F1.1.2
---------------	----------------------

Tabella 3: Tracciamento Classi - Requisiti

A.3 Tracciamento Modulo - Test

Metodo	Test
Sivodim::Drama::Model::Screenplay::Screenplay- ::addChapter(Chapter chapter)	TU7
Sivodim::Drama::Model::Screenplay- ::Screenplay::addCharacter(Character character)	TU8
Sivodim::Drama::Model::Screenplay::Screenplay::export()	TU5
Sivodim::Drama::Model::Screenplay::Screenplay- ::getChapter(String title)	TU38
Sivodim::Drama::Model::Screenplay::Screenplay- ::getChapterIterator()	
Sivodim::Drama::Model::Screenplay- ::Screenplay::getCharacterByName(String name)	TU38
Sivodim::Drama::Model::Screenplay::Screenplay::getCharacters()	TU10
Sivodim::Drama::Model::Screenplay::Screenplay::getTitle()	TU38
Sivodim::Drama::Model::Screenplay- ::Screenplay::importCharacters(Screenplay screenplay)	TU10
Sivodim::Drama::Model::Screenplay- ::Screenplay::removeCharacter(Character character)	TU9
Sivodim::Drama::Model::Screenplay::Screenplay::share()	TU6
Sivodim::Drama::Model::Screenplay- ::ScreenplayImpl::ScreenplayImpl(String title)	
Sivodim::Drama::Model::Screenplay::ScreenplayImpl- ::loadScreenplay(String screenplayTitle, Context context)	
Sivodim::Drama::Model::Screenplay::ScreenplayImpl- ::saveScreenplay(Screenplay screenplay, Context context)	TU4
Sivodim::Drama::Model::Screenplay- ::ScreenplayImpl::addChapter(Chapter chapter)	TU7

Sivodim::Drama::Model::Screenplay::ScreenplayImpl- ::getChapterIterator()	TU38
Sivodim::Drama::Model::Screenplay::ScreenplayImpl- ::export(String type)	TU5
Sivodim::Drama::Model::Screenplay::ScreenplayImpl::getTitle()	TU38
Sivodim::Drama::Model::Screenplay::ScreenplayImpl- ::share(String where)	TU6
Sivodim::Drama::Model::Screenplay::ScreenplayImpl- ::getCharacters()	TU10
Sivodim::Drama::Model::Screenplay- ::ScreenplayImpl::importCharacters(Screenplay screenplay)	TU10
Sivodim::Drama::Model::Screenplay- ::ScreenplayImpl::addCharacter(Character character)	TU8
Sivodim::Drama::Model::Screenplay- ::ScreenplayImpl::removeCharacter(Character character)	TU9
Sivodim::Drama::Model::Screenplay::ScreenplayImpl- ::getChapter(String title)	TU38
Sivodim::Drama::Model::Screenplay- ::ScreenplayImpl::getCharacterByName(String name)	TU38
Sivodim::Drama::Model::Character::Character::clone()	
Sivodim::Drama::Model::Character::Character::getAvatar()	TU44
Sivodim::Drama::Model::Character::Character::getName()	TU44
Sivodim::Drama::Model::Character::Character::getVoiceld()	TU44
Sivodim::Drama::Model::Character::Character::setAvatar(Avatar avatar)	TU43
Sivodim::Drama::Model::Character::Character::setName(String name)	TU43
Sivodim::Drama::Model::Character::Character::setVoice(String voicelD)	TU43
Sivodim::Drama::Model::Character- ::CharacterImpl::CharacterImpl(CharacterBuilder builder)	

Sivodim::Drama::Model::Character::CharacterImpl::clone()	
Sivodim::Drama::Model::Character::CharacterImpl- ::setAvatar(Avatar avatar)	TU43
Sivodim::Drama::Model::Character::CharacterImpl- ::setVoice(String voiceID)	TU43
Sivodim::Drama::Model::Character::CharacterImpl- ::setName(String name)	TU43
Sivodim::Drama::Model::Character::CharacterImpl::getName()	TU44
Sivodim::Drama::Model::Character::CharacterImpl::getAvatar()	TU44
Sivodim::Drama::Model::Character::CharacterImpl::getVoiceID()	TU44
Sivodim::Drama::Model::Character::CharacterBuilder::getResults()	
Sivodim::Drama::Model::Character::CharacterBuilder- ::setAvatar(Avatar avatar)	TU43
Sivodim::Drama::Model::Character::CharacterBuilder- ::setVoice(String voiceID)	TU43
Sivodim::Drama::Model::Character::CharacterBuilder- ::setName(String name)	TU43
Sivodim::Drama::Model::Character::CharacterBuilderImpl- ::CharacterBuilderImpl()	
Sivodim::Drama::Model::Character- ::CharacterBuilderImpl::setName(String name)	TU43
Sivodim::Drama::Model::Character- ::CharacterBuilderImpl::setAvatar(Avatar avatar)	TU43
Sivodim::Drama::Model::Character::CharacterBuilderImpl- ::setVoice(String voiceID)	TU43
Sivodim::Drama::Model::Utilities::Sound::Sound(String path)	
Sivodim::Drama::Model::Utilities::Sound::getAudio()	
Sivodim::Drama::Model::Utilities::Sound::pause()	
Sivodim::Drama::Model::Utilities::Sound::stop()	
Sivodim::Drama::Model::Utilities::Sound::play()	TU9
Sivodim::Drama::Model::Utilities::Sound::maxSize()	
Sivodim::Drama::Model::Utilities::Sound::isDefined()	

Sivodim::Drama::Model::Utilities::SoundFx::SoundFx(String path)	
Sivodim::Drama::Model::Utilities::SoundFx::maxSize()	
Sivodim::Drama::Model::Utilities::SpeechSound- ::SpeechSound(String path)	
Sivodim::Drama::Model::Utilities::SpeechSound::maxSize()	
Sivodim::Drama::Model::Utilities::Soundtrack::Soundtrack(String path)	
Sivodim::Drama::Model::Utilities::Soundtrack::maxSize()	
Sivodim::Drama::Model::Utilities::Image::Image(String path)	
Sivodim::Drama::Model::Utilities::Image::maxSize()	
Sivodim::Drama::Model::Utilities::Background::Backgroud(String path)	
Sivodim::Drama::Model::Utilities::Background::maxSize()	
Sivodim::Drama::Model::Utilities::Avatar::Avatar(String path)	
Sivodim::Drama::Model::Utilities::Avatar::maxSize()	
Sivodim::Drama::Model::Chapter::Chapter::getSpeechIterator()	TU27
Sivodim::Drama::Model::Chapter::Chapter::getTitle()	TU36
Sivodim::Drama::Model::Chapter::Chapter- ::setSoundtrack(Soundtrack soundtrack)	TU29
Sivodim::Drama::Model::Chapter::Chapter::setTitle(String title)	TU34
Sivodim::Drama::Model::Chapter::Chapter- ::setBackground(Background background)	TU31
Sivodim::Drama::Model::Chapter::Chapter::addSpeech(Speech speech)	TU11
Sivodim::Drama::Model::Chapter::Chapter- ::deleteSpeech(Speech speech)	TU33
Sivodim::Drama::Model::Chapter::Chapter::deleteBackground()	TU32
Sivodim::Drama::Model::Chapter::Chapter::deleteSoundtrack()	TU30
Sivodim::Drama::Model::Chapter::Chapter- ::moveUpSpeech(ListIterator<Speech> iterator)	TU28
Sivodim::Drama::Model::Chapter::Chapter- ::moveDownSpeech(ListIterator<Speech> iterator)	TU28

Sivodim::Drama::Model::Chapter::ChapterImpl::getTitle()	TU36
Sivodim::Drama::Model::Chapter::ChapterImpl- ::ChapterImpl(ChapterBuilder builder)	TU34
Sivodim::Drama::Model::Chapter::ChapterImpl- ::getSpeechIterator()	TU36
Sivodim::Drama::Model::Chapter::ChapterImpl- ::moveUpSpeech(ListIterator<Speech> iterator)	TU28
Sivodim::Drama::Model::Chapter::ChapterImpl- ::moveDownSpeech(ListIterator<Speech> iterator)	TU28
Sivodim::Drama::Model::Chapter::ChapterImpl- ::addSpeech(Speech speech)	TU11
Sivodim::Drama::Model::Chapter::ChapterImpl- ::setTitle(String title)	TU34
Sivodim::Drama::Model::Chapter::ChapterImpl- ::setBackground(Background background)	TU31
Sivodim::Drama::Model::Chapter::ChapterImpl- ::setSoundtrack(Soundtrack soundtrack)	TU29
Sivodim::Drama::Model::Chapter::ChapterImpl- ::deleteSpeech(Speech speech)	TU33
Sivodim::Drama::Model::Chapter::ChapterImpl- ::deleteBackground()	TU32
Sivodim::Drama::Model::Chapter::ChapterImpl- ::deleteSoundtrack()	TU30
Sivodim::Drama::Model::Chapter::ChapterBuilder- ::setTitle(String title)	TU34
Sivodim::Drama::Model::Chapter::ChapterBuilder- ::setBackground(Background background)	TU31
Sivodim::Drama::Model::Chapter::ChapterBuilder- ::setSoundFx(SoundFx soundFx)	TU35
Sivodim::Drama::Model::Chapter::ChapterBuilder- ::setSoundtrack(Soundtrack soundtrack)	TU29
Sivodim::Drama::Model::Chapter::ChapterBuilder::getResut()	
Sivodim::Drama::Model::Chapter::ChapterBuilderImpl- ::ChapterBuilderImpl()	TU34

Sivodim::Drama::Model::Chapter::ChapterBuilderImpl- ::setTitle(String title)	TU35
Sivodim::Drama::Model::Chapter::ChapterBuilderImpl- ::setBackground(Background background)	TU31
Sivodim::Drama::Model::Chapter::ChapterBuilderImpl- ::setSoundtrack(Soundtrack soundtrack)	TU29
Sivodim::Drama::Model::Chapter::ChapterBuilderImpl- ::setSoundFx(SoundFx soundFx)	TU35
Sivodim::Drama::Model::Chapter::ChapterBuilderImpl- ::getResut()	
Sivodim::Drama::Model::Chapter::Speech::setText(String text)	TU35
Sivodim::Drama::Model::Chapter::Speech::setEmotion(String emotionID)	TU35
Sivodim::Drama::Model::Chapter::Speech- ::setCharacter(Character character)	TU35
Sivodim::Drama::Model::Chapter::Speech::setSoundFx(SoundFx soundFx)	TU35
Sivodim::Drama::Model::Chapter::Speech::getText()	TU37
Sivodim::Drama::Model::Chapter::Speech::getEmotion()	TU37
Sivodim::Drama::Model::Chapter::Speech::getCharacter()	TU37
Sivodim::Drama::Model::Chapter::Speech::getAudio()	TU37
Sivodim::Drama::Model::Chapter::SpeechImpl- ::SpeechImpl(SpeechBuilder builder)	
Sivodim::Drama::Model::Chapter::SpeechImpl::setText(String text)	TU35
Sivodim::Drama::Model::Chapter::SpeechImpl- ::setEmotion(String emotionID)	TU35
Sivodim::Drama::Model::Chapter::SpeechImpl- ::setCharacter(Character character)	TU35
Sivodim::Drama::Model::Chapter::SpeechImpl- ::setSoundFx(SoundFx soundFx)	TU35
Sivodim::Drama::Model::Chapter::SpeechImpl::getText()	TU37
Sivodim::Drama::Model::Chapter::SpeechImpl::getEmotion()	TU37
Sivodim::Drama::Model::Chapter::SpeechImpl::getCharacter()	TU37

Sivodim::Drama::Model::Chapter::SpeechImpl::getAudio()	TU37
Sivodim::Drama::Model::Chapter::SpeechImpl- ::getEmotions(Context context)	TU37
Sivodim::Drama::Model::Chapter::SpeechImpl- ::getVoices(Context context)	
Sivodim::Drama::Presenter::HomePresenter- ::getTitlesAdapter(Context context)	TU1
Sivodim::Drama::Presenter::HomePresenter- ::getScreenplayTitles(Context context)	TU1
Sivodim::Drama::Presenter::HomePresenter- ::goToListChapter(Context context, String selected)	TU2
Sivodim::Drama::Presenter::HomePresenterImpl- ::HomePresenterImpl(HomeInterface homeInterface)	
Sivodim::Drama::Presenter::HomePresenterImpl- ::getScreenplayTitles(Context context)	TU1
Sivodim::Drama::Presenter::HomePresenterImpl- ::getTitlesAdapter(Context context)	TU1
Sivodim::Drama::Presenter::HomePresenterImpl- ::goToListChapter(Context context, String selected)	TU2
Sivodim::Drama::Presenter::ScreenplayPresenter- ::setActivity(ListChapterInterface listChapterInterface)	TU26
Sivodim::Drama::Presenter::ScreenplayPresenter- ::setActivity(NewChapterInterface newChapterInterface)	TU26
Sivodim::Drama::Presenter::ScreenplayPresenter- ::getTitlesAdapter(Context context, String screenplay)	
Sivodim::Drama::Presenter::ScreenplayPresenter- ::getScreenplayTitle()	TU38
Sivodim::Drama::Presenter::ScreenplayPresenter- ::getScreenplay()	TU38
Sivodim::Drama::Presenter::ScreenplayPresenter::export()	TU5
Sivodim::Drama::Presenter::ScreenplayPresenter::share()	TU6

Sivodim::Drama::Presenter::ScreenplayPresenter- ::newChapter(String title)	TU34
Sivodim::Drama::Presenter::ScreenplayPresenter- ::newScreenplay(String title, context)	TU3
Sivodim::Drama::Presenter::ScreenplayPresenter- ::moveUpChapter(String chapterTitle)	TU27
Sivodim::Drama::Presenter::ScreenplayPresenter- ::moveDownChapter(String chapterTitle)	TU27
Sivodim::Drama::Presenter::ScreenplayPresenter- ::save(Screenplay screenplay, context)	TU4
Sivodim::Drama::Presenter::ScreenplayPresenter- ::addCharacter(Character character)	TU8
Sivodim::Drama::Presenter::ScreenplayPresenter- ::importCharacters(String screenplay, context)	TU10
Sivodim::Drama::Presenter::ScreenplayPresenter- ::goToListSpeechesActivity(Context context, String selected)	TU26
Sivodim::Drama::Presenter::ScreenplayPresenter- ::goToListCharactersActivity(Context context)	TU26
Sivodim::Drama::Presenter::ScreenplayPresenter- ::goToEditChapterActivity(Context context, String selected)	TU26
Sivodim::Drama::Presenter::ScreenplayPresenter- ::goToNewChapterActivity(Context context)	TU26
Sivodim::Drama::Presenter::ScreenplayPresenter- ::goToNewCharacterActivity(Context context)	TU26
Sivodim::Drama::View::HomeActivity::onCreate(Bundle savedInstanceState)	
Sivodim::Drama::View::ListChapterActivity::onCreate(Bundle savedInstanceState)	
Sivodim::Drama::View::NewScreenplayActivity::onCreate(Bundle savedInstanceState)	

Sivodim::Drama::View::NewChapterActivity::onCreate(Bundle savedInstanceState)	
Sivodim::Drama::View::EditChapterActivity::onCreate(Bundle savedInstanceState)	
Sivodim::Drama::View::NewSpeechActivity::onCreate(Bundle savedInstanceState)	
Sivodim::Drama::View::EditSpeechActivity::onCreate(Bundle savedInstanceState)	
Sivodim::Drama::View::NewCharacterActivity::onCreate(Bundle savedInstanceState)	
Sivodim::Drama::View::ListCharacterActivity::onCreate(Bundle savedInstanceState)	
Sivodim::Drama::View::EditCharacterActivity::onCreate(Bundle savedInstanceState)	
Sivodim::Drama::Presenter::ScreenplayPresenterImpl- ::ScreenplayPresenterImpl(Screenplay screenplay)	
Sivodim::Drama::Presenter::ScreenplayPresenterImpl- ::setActivity(ListChapterInterface listChapterInterface)	TU26
Sivodim::Drama::Presenter::ScreenplayPresenterImpl- ::getScreenplay()	TU38
Sivodim::Drama::Presenter::ScreenplayPresenterImpl- ::getScreenplayTitle()	TU38
Sivodim::Drama::Presenter::ScreenplayPresenterImpl- ::getTitlesAdapter(Context context, String screenplay)	
Sivodim::Drama::Presenter::ScreenplayPresenterImpl- ::goToListSpeechesActivity(Context context, String selected)	TU26
Sivodim::Drama::Presenter::ScreenplayPresenterImpl- ::goToListCharactersActivity(Context context)	TU26
Sivodim::Drama::Presenter::ScreenplayPresenterImpl- ::goToEditChapterActivity(Context context, String selected)	TU26
Sivodim::Drama::Presenter::ScreenplayPresenterImpl- ::goToNewChapterActivity(Context context)	TU26

Sivodim::Drama::Presenter::ScreenplayPresenterImpl- ::goToNewCharacterActivity(Context context)	TU26
Sivodim::Drama::Presenter::ScreenplayPresenterImpl::export()	TU5
Sivodim::Drama::Presenter::ScreenplayPresenterImpl::share()	TU6
Sivodim::Drama::Presenter::ScreenplayPresenterImpl- ::newScreenplay(String title, Context context)	TU3
Sivodim::Drama::Presenter::ScreenplayPresenterImpl- ::addCharacter(Character character)	TU8
Sivodim::Drama::Presenter::ScreenplayPresenterImpl- ::importCharacters(String screenplay, Context context)	TU10
Sivodim::Drama::Presenter::ScreenplayPresenterImpl- ::newChapter(String title)	TU34
Sivodim::Drama::Presenter::ScreenplayPresenterImpl- ::moveUpChapter(String chapterTitle)	TU27
Sivodim::Drama::Presenter::ScreenplayPresenterImpl- ::moveDownChapter(String chapterTitle)	TU27
Sivodim::Drama::Presenter::ScreenplayPresenterImpl- ::save(Screenplay screenplay, Context context)	TU4
Sivodim::Drama::Presenter::ChapterPresenter- ::moveUpSpeech(Speech speech)	TU28
Sivodim::Drama::Presenter::ChapterPresenter- ::moveDownSpeech(Speech speech)	TU28
Sivodim::Drama::Presenter::ChapterPresenter- ::setActivity(NewSpeechInterface newSpeechInterface)	TU26
Sivodim::Drama::Presenter::ChapterPresenter- ::getSpeeches(Context context)	TU36
Sivodim::Drama::Presenter::ChapterPresenter- ::getCharactersAdapter(Context context)	
Sivodim::Drama::Presenter::ChapterPresenter::getChapterTitle()	TU36
Sivodim::Drama::Presenter::ChapterPresenter- ::getSpeechSelected()	
Sivodim::Drama::Presenter::ChapterPresenter- ::setChapterTitle(String title)	

Sivodim::Drama::Presenter::ChapterPresenter- ::setSpeechSelected(int position)	
Sivodim::Drama::Presenter::ChapterPresenter- ::newSpeech(String text, String characterName, String emotion)	TU45
Sivodim::Drama::Presenter::ChapterPresenter- ::deleteSpeech(Speech speech)	
Sivodim::Drama::Presenter::ChapterPresenter- ::goToEditSpeechActivity(Context context, Speech selected)	TU26
Sivodim::Drama::Presenter::ChapterPresenter- ::goToListCharactersActivity(Context context)	TU26
Sivodim::Drama::Presenter::ChapterPresenter- ::goToEditChapterActivity(Context context)	TU26
Sivodim::Drama::Presenter::ChapterPresenter- ::goToNewCharacterActivity(Context context)	TU26
Sivodim::Drama::Presenter::ChapterPresenter- ::goToNewSpeechActivity(Context context)	TU26
Sivodim::Drama::Presenter::ChapterPresenterImpl- ::ChapterPresenterImpl(Chapter chapter, CharacterContainer characterContainer, String projectName)	
Sivodim::Drama::Presenter::ChapterPresenterImpl- ::setActivity(ListSpeechesInterface listSpeechesInterface)	TU26
Sivodim::Drama::Presenter::ChapterPresenterImpl- ::getChapterTitle()	TU36
Sivodim::Drama::Presenter::ChapterPresenterImpl- ::getSpeeches()	TU36
Sivodim::Drama::Presenter::ChapterPresenterImpl- ::getCharactersAdapter(Context context)	
Sivodim::Drama::Presenter::ChapterPresenterImpl- ::getSpeechSelected()	
Sivodim::Drama::Presenter::ChapterPresenterImpl- ::setChapterTitle(String title)	

Sivodim::Drama::Presenter::ChapterPresenterImpl- ::setSpeechSelected(int position)	
Sivodim::Drama::Presenter::ChapterPresenterImpl- ::newSpeech(String text, String characterName, String emotion)	
Sivodim::Drama::Presenter::ChapterPresenterImpl- ::deleteSpeech(Speech speech)	
Sivodim::Drama::Presenter::ChapterPresenterImpl- ::goToEditSpeechActivity(Context context, Speech selected)	TU26
Sivodim::Drama::Presenter::ChapterPresenterImpl- ::goToListCharactersActivity(Context context)	TU26
Sivodim::Drama::Presenter::ChapterPresenterImpl- ::goToEditChapterActivity(Context context)	TU26
Sivodim::Drama::Presenter::ChapterPresenterImpl- ::goToNewCharacterActivity(Context context)	TU26
Sivodim::Drama::Presenter::ChapterPresenterImpl- ::goToNewSpeechActivity(Context context)	TU26
Sivodim::Drama::Presenter::ChapterPresenterImpl- ::moveUpSpeech(Speech speech)	TU27
Sivodim::Drama::Presenter::ChapterPresenterImpl- ::moveDownSpeech(Speech speech)	TU28
Sivodim::Libraries::Model::Mivoq::MivoqConnection- ::getResponse()	TU14
Sivodim::Libraries::Model::Mivoq::MivoqConnection- ::sendRequest(String text)	TU14
Sivodim::Libraries::Model::Mivoq::MivoqConnection- ::setEffects(String s)	
Sivodim::Libraries::Model::Mivoq::MivoqConnection- ::setLocale(String s)	
Sivodim::Libraries::Model::Mivoq::MivoqConnection- ::setQueue(RequestQueue rq)	
Sivodim::Libraries::Model::Mivoq::MivoqConnection- ::setVoiceGender(String s)	

Sivodim::Libraries::Model::Mivoq::MivoqConnection- ::setVoiceName(String s)	
Sivodim::Libraries::Model::Mivoq::AbstractFactory- ::createConnection()	TU49
Sivodim::Libraries::Model::Mivoq::MivoqConnectionFactory- ::MivoqConnectionFactory()	TU49
Sivodim::Libraries::Model::Mivoq::MivoqConnectionFactory- ::createConnection()	TU49
Sivodim::Libraries::Model::Mivoq::MivoqTTSSingleton- ::MivoqTTSSingleton()	
Sivodim::Libraries::Model::Mivoq::MivoqTTSSingleton- ::createVoice(String name, String gender, String myLanguage)	TU15
Sivodim::Libraries::Model::Mivoq::MivoqTTSSingleton- ::getInstance()	
Sivodim::Libraries::Model::Mivoq::MivoqTTSSingleton- ::getVoices()	TU50
Sivodim::Libraries::Model::Mivoq::MivoqTTSSingleton- ::hasContext()	
Sivodim::Libraries::Model::Mivoq::MivoqTTSSingleton- ::setContext(Context T)	
Sivodim::Libraries::Model::Mivoq::MivoqTTSSingleton- ::removeVoice(int index)	TU50
Sivodim::Libraries::Model::Mivoq::MivoqTTSSingleton- ::speak(MivoqVoice v, String text)	TU12
Sivodim::Libraries::Model::Mivoq::MivoqTTSSingleton- ::synthesizeToFile(String path, MivoqVoice v, String text)	TU13
Sivodim::Libraries::Model::Mivoq::MivoqTTSSingleton- ::synthesizeText(MivoqVoice v, String text)	TU14
Sivodim::Libraries::Model::Mivoq::MivoqConnectionImpl- ::getResponse()	TU14
Sivodim::Libraries::Model::Mivoq::MivoqConnectionImpl- ::sendRequest(String text)	TU14
Sivodim::Libraries::Model::Mivoq::MivoqConnectionImpl- ::setEffects(String s)	

Sivodim::Libraries::Model::Mivoq::MivoqConnectionImpl- ::setLocale(String s)	
Sivodim::Libraries::Model::Mivoq::MivoqConnectionImpl- ::setQueue(RequestQueue rq)	
Sivodim::Libraries::Model::Mivoq::MivoqConnectionImpl- ::setVoiceGender(String s)	
Sivodim::Libraries::Model::Mivoq::MivoqConnectionImpl- ::setVoiceName(String s)	
Sivodim::Libraries::Model::Mivoq::MivoqConnectionImpl- ::MivoqConnectionImpl()	TU49
Sivodim::Libraries::Model::Mivoq::MivoqTTSService- ::MivoqTTSService()	
Sivodim::Libraries::Model::Mivoq::MivoqTTSService- ::onGetLanguage()	
Sivodim::Libraries::Model::Mivoq::MivoqTTSService- ::onIsLanguageAvailable(String lang, String country, String variant)	
Sivodim::Libraries::Model::Mivoq::MivoqTTSService- ::onLoadLanguage(String lang, String country, String variant)	
Sivodim::Libraries::Model::Mivoq::MivoqTTSService::onStop()	
Sivodim::Libraries::Model::Mivoq::MivoqTTSService- ::onSynthesizeText(SynthesisRequest req, SynthesisCallback call)	
Sivodim::Drama::Presenter::SpeechPresenter- ::setActivity(EditSpeechInterface editSpeechInterface)	TU26
Sivodim::Drama::Presenter::SpeechPresenter::getSpeechText()	TU37
Sivodim::Drama::Presenter::SpeechPresenter- ::getSpeechEmotion()	TU37
Sivodim::Drama::Presenter::SpeechPresenter- ::getSpeechCharacter()	TU37
Sivodim::Drama::Presenter::SpeechPresenter- ::getScreenplayCharacters()	
Sivodim::Drama::Presenter::SpeechPresenter- ::setSpeechText(String text)	TU35
Sivodim::Drama::Presenter::SpeechPresenter- ::setSpeechEmotion(String emotion)	TU35

Sivodim::Drama::Presenter::SpeechPresenter- ::setSpeechCharacter(Character character)	TU35
Sivodim::Drama::Presenter::SpeechPresenterImpl- ::SpeechPresenterImpl(Speech speech, CharacterContainer characterContainer)	
Sivodim::Drama::Presenter::SpeechPresenterImpl- ::SpeechPresenterImpl(EditSpeechInterface editSpeechActivity, Speech speech)	
Sivodim::Drama::Presenter::SpeechPresenterImpl- ::setActivity(EditSpeechInterface editSpeechInterface)	TU26
Sivodim::Drama::Presenter::SpeechPresenterImpl- ::getSpeechText()	TU37
Sivodim::Drama::Presenter::SpeechPresenterImpl- ::getSpeechEmotion()	TU37
Sivodim::Drama::Presenter::SpeechPresenterImpl- ::getScreenplayCharacters()	
Sivodim::Drama::Presenter::SpeechPresenterImpl- ::getSpeechCharacter()	TU37
Sivodim::Drama::Presenter::SpeechPresenterImpl- ::setSpeechText(String text)	TU35
Sivodim::Drama::Presenter::SpeechPresenterImpl- ::setSpeechEmotion(String emotion)	TU35
Sivodim::Drama::Presenter::SpeechPresenterImpl- ::setSpeechCharacter(Character character)	TU35
Sivodim::Drama::Presenter::CharacterPresenter- ::setActivity(ListCharacterInterface listCharacterInterface)	TU26
Sivodim::Drama::Presenter::CharacterPresenter- ::setActivity(NewCharacterInterface newCharacterInterface)	TU26
Sivodim::Drama::Presenter::CharacterPresenter- ::setActivity(EditCharacterInterface editCharacterInterface)	TU26
Sivodim::Drama::Presenter::CharacterPresenter::getCharacter()	
Sivodim::Drama::Presenter::CharacterPresenter- ::getCharacterAdapter(Context context)	

Sivodim::Drama::Presenter::CharacterPresenter- ::getProjectName()	
Sivodim::Drama::Presenter::CharacterPresenter- ::newCharacter(String name, String voice, Avatar avatar)	
Sivodim::Drama::Presenter::CharacterPresenter- ::goToEditCharacterActivity(Context context, Character character)	TU26
Sivodim::Drama::Presenter::CharacterPresenter- ::goToNewCharacterActivity(Context context)	TU26
Sivodim::Drama::Presenter::CharacterPresenter- ::goToListCharacterActivity(Context context)	TU26
Sivodim::Drama::Presenter::CharacterPresenterImpl- ::CharacterPresenterImpl(CharacterContainer characterContainer, String projectName)	
Sivodim::Drama::Presenter::CharacterPresenterImpl- ::CharacterPresenterImpl(Character character, String projectName)	
Sivodim::Drama::Presenter::CharacterPresenterImpl- ::CharacterPresenterImpl(NewCharacterInterface characterInterface)	
Sivodim::Drama::Presenter::CharacterPresenterImpl- ::setActivity(ListCharacterInterface listCharacterInterface)	TU26
Sivodim::Drama::Presenter::CharacterPresenterImpl- ::setActivity(EditCharacterInterface editCharacterInterface)	TU26
Sivodim::Drama::Presenter::CharacterPresenterImpl- ::setActivity(NewCharacterInterface newCharacterInterface)	TU26
Sivodim::Drama::Presenter::CharacterPresenterImpl- ::getCharacterArrayAdapter(Context context)	
Sivodim::Drama::Presenter::CharacterPresenterImpl- ::getCharacter()	
Sivodim::Drama::Presenter::CharacterPresenterImpl- ::getProjectName()	

Sivodim::Drama::Presenter::CharacterPresenterImpl- ::newCharacter(String name, String voice, Avatar avatar)	TU8
Sivodim::Drama::Presenter::CharacterPresenterImpl- ::goToEditCharacterActivity(Context context, Character selected)	TU26
Sivodim::Drama::Presenter::CharacterPresenterImpl- ::goToNewCharacterActivity(Context context)	TU26
Sivodim::Drama::Presenter::CharacterPresenterImpl- ::goToListCharacterActivity(Context context)	TU26
Sivodim::Libraries::Model::Voice::Effect::getName()	TU16
Sivodim::Libraries::Model::Voice::Effect::getValue()	TU16
Sivodim::Libraries::Model::Voice::Effect::setValue(String f)	TU18
Sivodim::Libraries::Model::Voice::Effect::toString()	TU16
Sivodim::Libraries::Model::Voice::EffectImpl::getName()	TU16
Sivodim::Libraries::Model::Voice::EffectImpl::getValue()	TU16
Sivodim::Libraries::Model::Voice::EffectImpl::setValue(String f)	TU18
Sivodim::Libraries::Model::Voice::EffectImpl::toString()	TU16
Sivodim::Libraries::Model::Voice::EffectImpl::EffectImpl(String name)	TU16
Sivodim::Libraries::Model::Voice::Emotion::Emotion(String s)	TU17
Sivodim::Libraries::Model::Voice::Emotion::toString()	TU17
Sivodim::Libraries::Model::Voice::Language::Language(String l)	TU19
Sivodim::Libraries::Model::Voice::Language::setLanguage(String l)	TU20
Sivodim::Libraries::Model::Voice::Language::setState(String s)	TU20
Sivodim::Libraries::Model::Voice::Language::setModifier(String m)	TU20
Sivodim::Libraries::Model::Voice::Language::toString()	TU19
Sivodim::Libraries::Model::Voice::MivoqVoice::MivoqVoice(String name, String myVoiceName, Language locale)	TU15
Sivodim::Libraries::Model::Voice::MivoqVoice::getEffects()	TU46

Sivodim::Libraries::Model::Voice::MivoqVoice- ::getEncodedName(String myLang)	gen, String	TU46
Sivodim::Libraries::Model::Voice::MivoqVoice::getGender()		TU46
Sivodim::Libraries::Model::Voice::MivoqVoice::getLanguage()		TU46
Sivodim::Libraries::Model::Voice::MivoqVoice::getName()		TU46
Sivodim::Libraries::Model::Voice::MivoqVoice- ::getSampleText(String myLang)		TU22
Sivodim::Libraries::Model::Voice::MivoqVoice::getStringEffects()		TU46
Sivodim::Libraries::Model::Voice::MivoqVoice::getVoiceName()		TU46
Sivodim::Libraries::Model::Voice::MivoqVoice::removeEffect(int index)		TU47
Sivodim::Libraries::Model::Voice::MivoqVoice::setEffect(Effect e)		TU21
Sivodim::Libraries::Model::Voice::MivoqVoice- ::setEmotion(Emotion e)		TU21
Sivodim::Libraries::Model::Voice::MivoqVoice- ::setFemaleDe(boolean b)		TU21
Sivodim::Libraries::Model::Voice::MivoqVoice::setGender(String g)		TU21
Sivodim::Libraries::Model::Voice::MivoqVoice- ::setGenderLanguage(String myLang)	gen, String	TU21
Sivodim::Libraries::Model::Voice::MivoqVoice::setName(String n)		TU21
Sivodim::Libraries::Model::EngineManager::Engine- ::createVoice(String name, String gender, String myLanguage)		TU15
Sivodim::Libraries::Model::EngineManager::Engine::getVoices()		TU51
Sivodim::Libraries::Model::EngineManager::Engine- ::removeVoice(int index)		TU50
Sivodim::Libraries::Model::EngineManager::Engine::speak(String voiceID, String text)		TU12
Sivodim::Libraries::Model::EngineManager::Engine- ::synthesizeToFile(String path, String voiceID, String myEmotion, String text, Engine.listener myListner)		TU13
Sivodim::Drama::View::ListSpeechesActivity- ::ListSpeechesActivity()		

Sivodim::Drama::View::ListSpeechesActivity- ::setPresenter(ChapterPresenter chapterPresenter)	
Sivodim::Drama::Model::Character::CharacterContainer- ::CharacterContainer()	
Sivodim::Drama::Model::Character::CharacterContainer- ::CharacterContainer(ArrayList<Character> characters)	
Sivodim::Drama::Model::Character::CharacterContainer- ::cloneList()	
Sivodim::Drama::Model::Character::CharacterContainer::clone()	
Sivodim::Drama::Model::Character::CharacterContainer- ::iterator()	
Sivodim::Drama::Model::Character- ::CharacterContainer::addCharacter(Character character)	TU8
Sivodim::Drama::Model::Character- ::CharacterContainer::removeCharacter(Character character)	
Sivodim::Drama::Model::Character- ::CharacterContainer::getCharacterByName(String name)	
Sivodim::Drama::Model::Screenplay- ::FfmpegConnector::FfmpegConnector(Context context)	
Sivodim::Drama::Model::Screenplay::FfmpegConnector::exec()	TU5
Sivodim::Drama::Model::Screenplay::FfmpegConnector- ::getCommand()	TU5
Sivodim::Drama::Model::Screenplay::Mp3Converter- ::Mp3Converter(Context context, File fileToConvert, File destination)	TU39
Sivodim::Drama::Model::Screenplay::Mp3Converter::setFile(File fileToConvert)	TU39
Sivodim::Drama::Model::Screenplay::Mp3Converter- ::setDestination(File destination)	TU39
Sivodim::Drama::Model::Screenplay::Mp3Converter- ::getCommand()	TU5

Sivodim::Drama::Model::Screenplay::AudioConcatenator- ::AudioConcatenator(Context context, File destination)	TU40
Sivodim::Drama::Model::Screenplay::AudioConcatenator- ::AudioConcatenator(Context context, Vector<File> files, File destination)	TU40
Sivodim::Drama::Model::Screenplay::AudioConcatenator- ::addFile(File file)	TU40
Sivodim::Drama::Model::Screenplay- ::AudioConcatenator::setDestination(File destination)	TU40
Sivodim::Drama::Model::Screenplay::AudioConcatenator- ::createFileList()	TU40
Sivodim::Drama::Model::Screenplay::AudioConcatenator- ::getCommand()	TU5
Sivodim::Drama::Model::Screenplay::AudioMixer- ::AudioMixer(Context context, File primaryFile, File secondaryFile, File destination)	TU41
Sivodim::Drama::Model::Screenplay::AudioMixer- ::setPrimaryFile(File primaryFile)	TU41
Sivodim::Drama::Model::Screenplay::AudioMixer:: setSecondaryFile(File secondaryFile)	TU41
Sivodim::Drama::Model::Screenplay::AudioMixer- ::setDestination(File destination)	TU41
Sivodim::Drama::Model::Screenplay::AudioMixer::getCommand()	TU5
Sivodim::Drama::Model::Chapter::SpeechBuilderImpl- ::SpeechBuilderImpl()	
Sivodim::Drama::Model::Chapter::SpeechBuilderImpl- ::setText(String text)	TU35
Sivodim::Drama::Model::Chapter::SpeechBuilderImpl- ::setEmotion(String emotionID)	TU35
Sivodim::Drama::Model::Chapter::SpeechBuilderImpl- ::setCharacter(Character character)	TU35
Sivodim::Drama::Model::Chapter::SpeechBuilderImpl- ::setSoundFX(SoundFx soundFx)	TU35
Sivodim::Drama::Model::Chapter::SpeechBuilderImpl::build()	

Sivodim::Drama::Presenter::CharacterArrayAdapter- ::CharacterArrayAdapter(Context context, int resource)	
Sivodim::Drama::Presenter::CharacterArrayAdapter- ::add(Character object)	TU8
Sivodim::Drama::Presenter::CharacterArrayAdapter::getCount()	
Sivodim::Drama::Presenter::CharacterArrayAdapter::getItem(int index)	
Sivodim::Drama::Presenter::CharacterArrayAdapter::getView(int position, View convertView, ViewGroup parent)	
Sivodim::Drama::Presenter::SpeechArrayAdapter- ::SpeechArrayAdapter(Context context, int resource)	
Sivodim::Drama::Presenter::SpeechArrayAdapter::add(Speech object)	TU11
Sivodim::Drama::Presenter::SpeechArrayAdapter::getCount()	
Sivodim::Drama::Presenter::SpeechArrayAdapter::getItem(int index)	
Sivodim::Drama::Presenter::SpeechArrayAdapter- ::setSpeechSelected(int position)	
Sivodim::Drama::Presenter::SpeechArrayAdapter::getView(int position, View convertView, ViewGroup parent)	
Sivodim::Libraries::Presenter::HomePresenterImpl- ::HomePresenterImpl(Context context)	TU23
Sivodim::Libraries::Presenter::HomePresenterImpl- ::goToNewVoiceActivity(Context context)	TU23
Sivodim::Libraries::Presenter::HomePresenterImpl- ::goToVoiceListActivity(Context context)	TU23
Sivodim::Libraries::Presenter::HomePresenter- ::goToVoiceListActivity(Context context)	TU23
Sivodim::Libraries::Presenter::HomePresenter- ::goToNewVoiceActivity(Context context)	TU23
Sivodim::Libraries::Presenter::VoiceListPresenter- ::goUpEditVoiceActivity(Context context, MivoqVoice mivoqVoice)	TU23

Sivodim::Libraries::Presenter::VoiceListPresenter- ::setActivity(VoiceListActivityInterface voiceListActivityInterface)	TU23
Sivodim::Libraries::Presenter::VoiceListPresenter- ::createMivoqVoice(String s)	TU15
Sivodim::Libraries::Presenter::VoiceListPresenter- ::loadVoiceNames(Context context)	TU25
Sivodim::Libraries::Presenter::VoiceListPresenter- ::getVoicesAdapter(Context context)	TU25
Sivodim::Libraries::Presenter::VoiceListPresenterImpl- ::VoiceListPresenterImpl(Engine engine)	TU23
Sivodim::Libraries::Presenter::VoiceListPresenter- ::goUpEditVoiceActivity(Context context, MivoqVoice mivoqVoice)	TU23
Sivodim::Libraries::Presenter::VoiceListPresenterImpl- ::setActivity(VoiceListActivityInterface voiceListActivityInterface)	TU23
Sivodim::Libraries::Presenter::VoiceListPresenterImpl- ::createMivoqVoice(String s)	TU15
Sivodim::Libraries::Presenter::VoiceListPresenterImpl- ::loadVoiceNames(Context context)	TU25
Sivodim::Libraries::Presenter::VoiceListPresenterImpl- ::getVoicesAdapter(Context context)	TU25
Sivodim::Libraries::Presenter::VoicePresenterImpl- ::VoicePresenterImpl(Engine engine)	TU23
Sivodim::Libraries::Presenter::VoicePresenterImpl- ::VoicePresenterImpl(Mivoq mivoq)	TU23
Sivodim::Libraries::Presenter::VoicePresenterImpl- ::setActivity(NewVoiceActivityInterface newVoiceActivityInterface)	TU23
Sivodim::Libraries::Presenter::VoicePresenterImpl- ::setActivity(EditVoiceActivityInterface newVoiceActivityInterface)	TU23
Sivodim::Libraries::Presenter::VoicePresenterImpl- ::loadVoiceGender(Context context)	TU24
Sivodim::Libraries::Presenter::VoicePresenterImpl- ::getGenderAdapter(Context context)	TU46

Sivodim::Libraries::Presenter::VoicePresenterImpl- ::loadVoiceLanguage(Context context)	TU24
Sivodim::Libraries::Presenter::VoicePresenterImpl- ::getLanguageAdapter(Context context)	TU46
Sivodim::Libraries::Presenter::VoicePresenterImpl- ::getVoiceName()	TU46
Sivodim::Libraries::Presenter::VoicePresenter- ::setActivity(NewVoiceActivityInterface newVoiceActivityInterface)	TU23
Sivodim::Libraries::Presenter::VoicePresenter- ::setActivity(EditVoiceActivityInterface newVoiceActivityInterface)	TU23
Sivodim::Libraries::Presenter::VoicePresenter- ::loadVoiceGender(Context context)	TU24
Sivodim::Libraries::Presenter::VoicePresenter- ::getGenderAdapter(Context context)	TU46
Sivodim::Libraries::Presenter::VoicePresenter- ::loadVoiceLanguage(Context context)	TU24
Sivodim::Libraries::Presenter::VoicePresenter- ::getLanguageAdapter(Context context)	TU46
Sivodim::Libraries::Presenter::VoicePresenter::getVoiceName()	TU46
Sivodim::Drama::Model::Screenplay::ExportAlgorithm- ::ExportAlgorithm()	TU5
Sivodim::Drama::Model::Screenplay- ::ExportAlgorithm::setScreenplay(Screenplay screenplay)	TU5
Sivodim::Drama::Model::Screenplay::ExportAlgorithm::export()	TU5
Sivodim::Drama::Model::Screenplay::AudioExport- ::AudioExport()	TU5
Sivodim::Drama::Model::Screenplay::AudioExport- ::concatenateSpeeches()	TU40
Sivodim::Drama::Model::Screenplay::AudioExport- ::addSoundtrack()	
Sivodim::Drama::Model::Screenplay::AudioExport- ::finalizeExport()	TU5
Sivodim::Drama::Model::Screenplay::AudioExport::export()	TU5

Sivodim::Drama::Model::Screenplay::VideoExport::VideoExport()	TU5
Sivodim::Drama::Model::Screenplay::VideoExport::exportAudio()	TU5
Sivodim::Drama::Model::Screenplay::VideoExport- ::concatenateImages()	TU42
Sivodim::Drama::Model::Screenplay::VideoExport- ::finalizeExport()	TU5
Sivodim::Drama::Model::Screenplay::VideoExport::export()	TU5
Sivodim::Drama::Model::Utilities::XMLParser::XMLParser()	
Sivodim::Drama::Model::Utilities::XMLParser::getParsedData()	TU4
Sivodim::Drama::Model::Utilities::XMLParser- ::parseCharacter(Node node)	TU4
Sivodim::Drama::Model::Utilities::XMLParser- ::parseChapter(Node node)	TU4
Sivodim::Drama::Model::Utilities::XMLParser::parseXml(File file)	TU4
Sivodim::Drama::Model::Utilities::XMLParser::saveXML(File file, Screenplay screenplay)	TU4
Sivodim::Libraries::Model::Mivoq::AudioRequest- ::AudioRequest(int post, String mUrl, Map<String, String> params, Response.Listener<byte[]> listener, Response.ErrorListener errorListener)	TU48
Sivodim::Libraries::Model::Mivoq::AudioRequest- ::deliverResponse(byte[] response)	TU14
Sivodim::Libraries::Model::Mivoq::AudioRequest::getHeaders()	TU14
Sivodim::Libraries::Model::Mivoq::AudioRequest::getParams()	TU14
Sivodim::Libraries::Model::Mivoq::AudioRequest- ::parseNetworkResponse(NetworkResponse response)	TU14
Sivodim::Libraries::Model::EngineManager::EnginelImpl- ::EnginelImpl(Context c)	
Sivodim::Libraries::Model::EngineManager::EnginelImpl- ::createVoice(String name, String gender, String myLanguage)	TU15
Sivodim::Libraries::Model::EngineManager::EnginelImpl- ::getVoices()	TU51
Sivodim::Libraries::Model::EngineManager::EnginelImpl- ::removeVoice(int index)	TU50

Sivodim::Libraries::Model::EngineManager- ::EnginelImpl::speak(String text)	voicelD, String	TU12
Sivodim::Libraries::Model::EngineManager::EnginelImpl- ::synthesizeToFile(String path, String voicelD, String myEmotion, String text, Engine.listener myListener)		TU13

Tabella 4: Tracciamento Modulo - Test