

Table of Contents

NPCEngine

NPCEngineException

ResultFuture<ReturnType>

NPCEngine.API

Chatbot<ContextType>

SemanticQuery

SpeechToText

TextToSpeech

NPCEngine.Components

AbstractDialogueSystem

AbstractSpeechToText

ChatLine

ColliderLocationTrigger

DialogueUI

DictationRecognizerSTT

FantasyChatbotContext

NonPlayerCharacter

NPCEngineSTT

PlayerCharacter

NPCEngine.Server

NPCEngineServer

RPCRequestMessage<ParametersType>

RPCResponseError

RPCResponseMessage<ReturnType>

NPCEngine.Utility

AudioSourceQueue

Singleton<T>

Namespace NPCEngine

Classes

[NPCEngineException](#)

[ResultFuture<ReturnType>](#)

Class NPCEngineException

Inheritance

System.Object
System.Exception
NPCEngineException

Namespace: [NPCEngine](#)
Assembly: cs.temp.dll.dll

Syntax

```
public class NPCEngineException : Exception, ISerializable, _Exception
```

Constructors

NPCEngineException()

Declaration

```
public NPCEngineException()
```

NPCEngineException(SerializationInfo, StreamingContext)

Declaration

```
protected NPCEngineException(SerializationInfo info, StreamingContext context)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Runtime.Serialization.SerializationInfo	info	
System.Runtime.Serialization.StreamingContext	context	

NPCEngineException(String)

Declaration

```
public NPCEngineException(string message)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	message	

NPCEngineException(String, Exception)

Declaration

```
public NPCEngineException(string message, Exception inner)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	message	
System.Exception	inner	

Class ResultFuture<ReturnType>

Inheritance

System.Object

ResultFuture<ReturnType>

Namespace: [NPCEngine](#)

Assembly: cs.temp.dll.dll

Syntax

```
public class ResultFuture<ReturnType>
```

Type Parameters

NAME	DESCRIPTION
ReturnType	

Properties

Error

Declaration

```
public NPCEngineException Error { get; }
```

Property Value

TYPE	DESCRIPTION
NPCEngineException	

Result

Declaration

```
public ReturnType Result { get; }
```

Property Value

TYPE	DESCRIPTION
ReturnType	

ResultReady

Declaration

```
public bool ResultReady { get; }
```

Property Value

TYPE	DESCRIPTION
System.Boolean	

Methods

ErrorCallback(NPCEngineException)

Declaration

```
public void errorCallback(NPCEngineException error)
```

Parameters

TYPE	NAME	DESCRIPTION
NPCEngineException	error	

ResultFinishedCallback(ReturnType)

Declaration

```
public void ResultFinishedCallback(ReturnType result)
```

Parameters

TYPE	NAME	DESCRIPTION
ReturnType	result	

Namespace NPCEngine.API

Classes

[Chatbot<ContextType>](#)

Static class `Chatbot` provides remote procedure calls to inference engine's chatbot model.

[SemanticQuery](#)

Static class `SemanticQuery` provides remote procedure calls to inference engine's Semantic similarity model. Prefer predefining queries via `PredefineQuery` and then using `QueryPredefined` instead of directly using `Query`

[SpeechToText](#)

Static class `SpeechToText` provides remote procedure calls to inference engine's speech to text module.

[TextToSpeech](#)

Static class `TextToSpeech` provides remote procedure calls to inference engine's text to speech module.

Class Chatbot<ContextType>

Static class `Chatbot` provides remote procedure calls to inference engine's chatbot model.

Inheritance

System.Object

Chatbot<ContextType>

Namespace: `NPCEngine.API`

Assembly: `cs.temp.dll.dll`

Syntax

```
public static class Chatbot<ContextType>
    where ContextType : new()
```

Type Parameters

NAME	DESCRIPTION
ContextType	

Methods

GenerateReply(ContextType, Single, Int32)

Declaration

```
public static ResultFuture<string> GenerateReply(ContextType context, float temperature = 0.8F, int topk = 5)
```

Parameters

TYPE	NAME	DESCRIPTION
ContextType	context	
System.Single	temperature	
System.Int32	topk	

Returns

TYPE	DESCRIPTION
<code>ResultFuture<System.String></code>	

GenerateReplyCoroutine(ContextType, Action<String>, Single, Int32)

Declaration

```
public static IEnumerable<string> GenerateReplyCoroutine(ContextType context, Action<string> outputCallback, float temperature = 0.8F, int topk = 5)
```

Parameters

TYPE	NAME	DESCRIPTION
ContextType	context	
Action<System.String>	outputCallback	

TYPE	NAME	DESCRIPTION
System.Single	temperature	
System.Int32	topk	

Returns

TYPE	DESCRIPTION
System.Collections.Generic.IEnumerator<System.String>	

GetPromptTemplate()

Declaration

```
public static ResultFuture<string> GetPromptTemplate()
```

Returns

TYPE	DESCRIPTION
ResultFuture<System.String>	

GetPromptTemplateCoroutine(Action<String>)

Declaration

```
public static IEnumerable GetPromptTemplateCoroutine(Action<string> outputCallback)
```

Parameters

TYPE	NAME	DESCRIPTION
Action<System.String>	outputCallback	

Returns

TYPE	DESCRIPTION
System.Collections.IEnumerator	

GetSpecialTokens()

Declaration

```
public static ResultFuture<Dictionary<string, string>> GetSpecialTokens()
```

Returns

TYPE	DESCRIPTION
ResultFuture<Dictionary<System.String, System.String> >	

GetSpecialTokensCoroutine(Action<Dictionary<String, String>>)

Declaration

```
public static IEnumerable<Dictionary<string, string>> GetSpecialTokensCoroutine(Action<Dictionary<string, string>> outputCallback)
```


Parameters

TYPE	NAME	DESCRIPTION
Action< Dictionary <System.String, System.String>>	outputCallback	

Returns

TYPE	DESCRIPTION
System.Collections.Generic.IEnumerator< Dictionary <System.String, System.String>>	

Class SemanticQuery

Static class `SemanticQuery` provides remote procedure calls to inference engine's Semantic similarity model. Prefer predefining queries via `PredefineQuery` and then using `QueryPredefined` instead of directly using `Query`

Inheritance

System.Object
SemanticQuery

Namespace: `NPCEngine.API`

Assembly: `cs.temp.dll.dll`

Syntax

```
public static class SemanticQuery
```

Methods

Cache(List<String>)

Declaration

```
public static void Cache(List<string> queryIds)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Collections.Generic.List<System.String>	queryIds	

Compare(String, List<String>)

Declaration

```
public static ResultFuture<List<float>> Compare(string query, List<string> context)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	query	
System.Collections.Generic.List<System.String>	context	

Returns

TYPE	DESCRIPTION
<code>ResultFuture<System.Collections.Generic.List<System.Single>></code>	

CompareCoroutine(String, List<String>, Action<List<Single>>)

Declaration

```
public static IEnumerable CompareCoroutine(string query, List<string> context, Action<List<float>> outputCallback)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	query	
System.Collections.Generic.List<System.String>	context	
Action<System.Collections.Generic.List<System.Single> >	outputCallback	

Returns

TYPE	DESCRIPTION
System.Collections.IEnumerator	

Class SpeechToText

Static class `SpeechToText` provides remote procedure calls to inference engine's speech to text module.

Inheritance

System.Object

SpeechToText

Namespace: `NPCEngine.API`

Assembly: `cs.temp.dll.dll`

Syntax

```
public static class SpeechToText
```

Methods

GetDevices()

Declaration

```
public static ResultFuture<List<string>> GetDevices()
```

Returns

TYPE	DESCRIPTION
<code>ResultFuture<System.Collections.Generic.List<System.String>></code>	

InitializeMicrophoneInput()

Declaration

```
public static IEnumerable InitializeMicrophoneInput()
```

Returns

TYPE	DESCRIPTION
<code>System.Collections.IEnumerator</code>	

Listen(String)

Declaration

```
public static ResultFuture<string> Listen(string context)
```

Parameters

TYPE	NAME	DESCRIPTION
<code>System.String</code>	<code>context</code>	

Returns

TYPE	DESCRIPTION
<code>ResultFuture<System.String></code>	

SetDevice(Int32)

Declaration

```
public static void SetDevice(int deviceId)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	deviceId	

Transcribe(List<Single>)

Declaration

```
public static ResultFuture<string> Transcribe(List<float> audio)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Collections.Generic.List<System.Single>	audio	

Returns

TYPE	DESCRIPTION
ResultFuture<System.String>	

Class TextToSpeech

Static class `TextToSpeech` provides remote procedure calls to inference engine's text to speech module.

Inheritance

System.Object

TextToSpeech

Namespace: `NPCEngine.API`

Assembly: `cs.temp.dll.dll`

Syntax

```
public static class TextToSpeech
```

Methods

GetNextResult()

Declaration

```
public static ResultFuture<List<float>> GetNextResult()
```

Returns

TYPE	DESCRIPTION
<code>ResultFuture<System.Collections.Generic.List<System.Single>></code>	

GetNextResultCoroutine(Action<List<Single>>)

Declaration

```
public static IEnumerator GetNextResultCoroutine(Action<List<float>> outputCallback)
```

Parameters

TYPE	NAME	DESCRIPTION
<code>Action<System.Collections.Generic.List<System.Single>></code>	<code>outputCallback</code>	

Returns

TYPE	DESCRIPTION
<code>System.Collections.IEnumerator</code>	

GetSpeakerIds()

Declaration

```
public static ResultFuture<List<string>> GetSpeakerIds()
```

Returns

TYPE	DESCRIPTION
<code>ResultFuture<System.Collections.Generic.List<System.String>></code>	

StartTTS(String, String, Int32)

Declaration

```
public static void StartTTS(string voiceId, string line, int n_chunks)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	voiceId	
System.String	line	
System.Int32	n_chunks	

Namespace NPCEngine.Components

Classes

[AbstractDialogueSystem](#)

Abstract class for dialogue system integration. Concrete implementations are used by NonPlayerCharacter component for scripted replies.

[AbstractSpeechToText](#)

Abstract class for speech to text system. Concrete implementations are used by PlayerCharacter component for receiving speech.

[ChatLine](#)

[ColliderLocationTrigger](#)

Simple collider trigger that sets location in PlayerCharacter

[DialogueUI](#)

Simple UI script that uses NonPlayerCharacter callbacks.

[DictationRecognizerSTT](#)

[FantasyChatbotContext](#)

[NonPlayerCharacter](#)

[NPCEngineSTT](#)

[PlayerCharacter](#)

Class AbstractDialogueSystem

Abstract class for dialogue system integration. Concrete implementations are used by NonPlayerCharacter component for scripted replies.

Inheritance

System.Object
AbstractDialogueSystem

Namespace: [NPCEngine.Components](#)

Assembly: cs.temp.dll.dll

Syntax

```
public abstract class AbstractDialogueSystem : MonoBehaviour
```

Methods

CurrentNodeIsPlayer()

Returns true if it's players turn in the dialogue.

Declaration

```
public abstract bool CurrentNodeIsPlayer()
```

Returns

TYPE	DESCRIPTION
System.Boolean	

CurrentNodeNPCAudio()

Get the current node's audio for the NPC line (Never called for player). Return null if no audio is set or if the node is player.

Declaration

```
public abstract AudioClip CurrentNodeNPCAudio()
```

Returns

TYPE	DESCRIPTION
AudioClip	

CurrentNodeNPCLine()

If the current node is NPC, return the NPC's line. else return empty string.

Declaration

```
public abstract string CurrentNodeNPCLine()
```

Returns

TYPE	DESCRIPTION
System.String	

CurrentNodeThreshold()

Get the semantic similarity threshold for the current node in the dialogue. Return -1 if no threshold is set.

Declaration

```
public abstract float CurrentNodeThreshold()
```

Returns

TYPE	DESCRIPTION
System.Single	

EndDialog()

Will be called when dialogue ends.

Declaration

```
public abstract void EndDialog()
```

GetCurrentNodeOptions()

Get the lines associated with the current node in the dialogue.

Declaration

```
public abstract List<string> GetCurrentNodeOptions()
```

Returns

TYPE	DESCRIPTION
System.Collections.Generic.List<System.String>	

GetCurrentNodeTopics()

Get the short abstract descriptions of the lines associated with the current node in the dialogue.

Declaration

```
public abstract List<string> GetCurrentNodeTopics()
```

Returns

TYPE	DESCRIPTION
System.Collections.Generic.List<System.String>	

Next()

Move to the next node in the dialogue.

Declaration

```
public abstract void Next()
```

SelectOption(Int32)

Select a line from the current node in the dialogue.

Declaration

```
public abstract void SelectOption(int optionId)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	optionId	

StartDialogue()

Will be called when dialogue starts.

Declaration

```
public abstract void StartDialogue()
```

Class AbstractSpeechToText

Abstract class for speech to text system. Concrete implementations are used by PlayerCharacter component for receiving speech.

Inheritance

System.Object
AbstractSpeechToText
[DictationRecognizerSTT](#)
[NPCEngineSTT](#)

Namespace: [NPCEngine.Components](#)
Assembly: cs.temp.dll.dll

Syntax

```
public abstract class AbstractSpeechToText : MonoBehaviour
```

Properties

Context

Consumed context to be used in the next STT request

Declaration

```
public string Context { protected get; set; }
```

Property Value

TYPE	DESCRIPTION
System.String	

Methods

SpeechRecognized(String)

Declaration

```
protected void SpeechRecognized(string result)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	result	

StartListening()

Called when system must start listening for speech.

Declaration

```
public abstract void StartListening()
```

StopListening()

Called when system should stop listening for speech.

Declaration

```
public abstract void StopListening()
```

Events

OnSpeechRecognitionFailed

Can be used to handle speech recogintion errors

Declaration

```
public event Action<string> OnSpeechRecognitionFailed
```

Event Type

TYPE	DESCRIPTION
Action<System.String>	

OnSpeechRecognized

PlayerCharacter subscribes to this event to get the Speech-to-Text result

Declaration

```
public event Action<string> OnSpeechRecognized
```

Event Type

TYPE	DESCRIPTION
Action<System.String>	

Class ChatLine

Inheritance

System.Object

ChatLine

Namespace: [NPCEngine.Components](#)

Assembly: cs.temp.dll.dll

Syntax

```
public class ChatLine
```

Fields

line

Declaration

```
public string line
```

Field Value

TYPE	DESCRIPTION
System.String	

speaker

Declaration

```
public string speaker
```

Field Value

TYPE	DESCRIPTION
System.String	

Class ColliderLocationTrigger

Simple collider trigger that sets location in PlayerCharacter

Inheritance

System.Object

ColliderLocationTrigger

Namespace: [NPCEngine.Components](#)

Assembly: cs.temp.dll.dll

Syntax

```
public class ColliderLocationTrigger : MonoBehaviour
```

Fields

locationCollider

Declaration

```
public Collider locationCollider
```

Field Value

TYPE	DESCRIPTION
Collider	

locationDescription

Declaration

```
public string locationDescription
```

Field Value

TYPE	DESCRIPTION
System.String	

locationName

Declaration

```
public string locationName
```

Field Value

TYPE	DESCRIPTION
System.String	

Class DialogueUI

Simple UI script that uses NonPlayerCharacter callbacks.

Inheritance

System.Object

DialogueUI

Namespace: [NPCEngine.Components](#)

Assembly: cs.temp.dll.dll

Syntax

```
public class DialogueUI : MonoBehaviour
```

Fields

dialogueHistory

Text field for the dialogue.

Declaration

```
protected Text dialogueHistory
```

Field Value

TYPE	DESCRIPTION
Text	

dialogueTopics

Text field for the topics of the possible dialogue options.

Declaration

```
protected Text dialogueTopics
```

Field Value

TYPE	DESCRIPTION
Text	

highlightColor

Declaration

```
public Color highlightColor
```

Field Value

TYPE	DESCRIPTION
Color	

lineLimit

Line limit for the text field.

Declaration


```
public int lineLimit
```

Field Value

TYPE	DESCRIPTION
System.Int32	

uiObject

GameObject that will be enabled, disabled and oriented towards the player.

Declaration

```
public GameObject uiObject
```

Field Value

TYPE	DESCRIPTION
GameObject	

usernameColor

Declaration

```
public Color usernameColor
```

Field Value

TYPE	DESCRIPTION
Color	

Methods

AddLine(ChatLine, Boolean)

Declaration

```
public void AddLine(ChatLine chatLine, bool highlight = false)
```

Parameters

TYPE	NAME	DESCRIPTION
ChatLine	chatLine	
System.Boolean	highlight	

Disable()

Declaration

```
public void Disable()
```

Enable()

Declaration

```
public void Enable()
```

UpdateTopics(List<String>)

Declaration

```
public void UpdateTopics(List<string> topics)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Collections.Generic.List<System.String>	topics	

Class DictationRecognizerSTT

Inheritance

System.Object
[AbstractSpeechToText](#)
DictationRecognizerSTT

Inherited Members

[AbstractSpeechToText.Context](#)
[AbstractSpeechToText.OnSpeechRecognized](#)
[AbstractSpeechToText.OnSpeechRecognitionFailed](#)
[AbstractSpeechToText.SpeechRecognized\(String\)](#)

Namespace: [NPCEngine.Components](#)
Assembly: cs.temp.dll.dll

Syntax

```
public class DictationRecognizerSTT : AbstractSpeechToText
```

Fields

debugLogs

Declaration

```
public bool debugLogs
```

Field Value

TYPE	DESCRIPTION
System.Boolean	

Properties

PolicyNotAccepted

Declaration

```
public bool PolicyNotAccepted { get; }
```

Property Value

TYPE	DESCRIPTION
System.Boolean	

RecognizerStarted

Declaration

```
public bool RecognizerStarted { get; }
```

Property Value

TYPE	DESCRIPTION
System.Boolean	

Methods

StartListening()

Declaration

```
public override void StartListening()
```

Overrides

[AbstractSpeechToText.StartListening\(\)](#)

StopListening()

Declaration

```
public override void StopListening()
```

Overrides

[AbstractSpeechToText.StopListening\(\)](#)

Class FantasyChatbotContext

Inheritance

System.Object

FantasyChatbotContext

Namespace: [NPCEngine.Components](#)

Assembly: cs.temp.dll.dll

Syntax

```
public class FantasyChatbotContext
```

Fields

history

Declaration

```
public List<ChatLine> history
```

Field Value

TYPE	DESCRIPTION
System.Collections.Generic.List< ChatLine >	

location

Declaration

```
public string location
```

Field Value

TYPE	DESCRIPTION
System.String	

location_name

Declaration

```
public string location_name
```

Field Value

TYPE	DESCRIPTION
System.String	

name

Declaration

```
public string name
```

Field Value

TYPE	DESCRIPTION
System.String	

other_name

Declaration

```
public string other_name
```

Field Value

TYPE	DESCRIPTION
System.String	

other_persona

Declaration

```
public string other_persona
```

Field Value

TYPE	DESCRIPTION
System.String	

persona

Declaration

```
public string persona
```

Field Value

TYPE	DESCRIPTION
System.String	

Class NonPlayerCharacter

Inheritance

System.Object

NonPlayerCharacter

Namespace: [NPCEngine.Components](#)

Assembly: cs.temp.dll.dll

Syntax

```
public class NonPlayerCharacter : MonoBehaviour
```

Fields

audioSourceQueue

Declaration

```
public AudioSourceQueue audioSourceQueue
```

Field Value

TYPE	DESCRIPTION
AudioSourceQueue	

characterName

Declaration

```
public string characterName
```

Field Value

TYPE	DESCRIPTION
System.String	

defaultThreshold

Declaration

```
public float defaultThreshold
```

Field Value

TYPE	DESCRIPTION
System.Single	

dialogueSystem

Declaration

```
public AbstractDialogueSystem dialogueSystem
```

Field Value

TYPE	DESCRIPTION
AbstractDialogueSystem	

history

Declaration

```
public List<ChatLine> history
```

Field Value

TYPE	DESCRIPTION
System.Collections.Generic.List< ChatLine >	

nChunksTextGeneration

Declaration

```
public int nChunksTextGeneration
```

Field Value

TYPE	DESCRIPTION
System.Int32	

OnDialogueEnd

Declaration

```
public UnityEvent OnDialogueEnd
```

Field Value

TYPE	DESCRIPTION
UnityEvent	

OnDialogueLine

Declaration

```
public UnityEvent<ChatLine, bool> OnDialogueLine
```

Field Value

TYPE	DESCRIPTION
UnityEvent< ChatLine , System.Boolean>	

OnDialogueStart

Declaration

```
public UnityEvent OnDialogueStart
```

Field Value

TYPE	DESCRIPTION
UnityEvent	

OnProcessingEnd

Declaration

```
public UnityEvent OnProcessingEnd
```

Field Value

TYPE	DESCRIPTION
UnityEvent	

OnProcessingStart

Declaration

```
public UnityEvent OnProcessingStart
```

Field Value

TYPE	DESCRIPTION
UnityEvent	

OnTopicHintsUpdate

Declaration

```
public UnityEvent<List<string>> OnTopicHintsUpdate
```

Field Value

TYPE	DESCRIPTION
UnityEvent<System.Collections.Generic.List<System.String>>	

persona

Declaration

```
public string persona
```

Field Value

TYPE	DESCRIPTION
System.String	

temperature

Declaration

```
public float temperature
```

Field Value

TYPE	DESCRIPTION
System.Single	

topK

Declaration

```
public int topK
```

Field Value

TYPE	DESCRIPTION
System.Int32	

voiceId

Declaration

```
public string voiceId
```

Field Value

TYPE	DESCRIPTION
System.String	

Properties

Initialized

Declaration

```
public bool Initialized { get; }
```

Property Value

TYPE	DESCRIPTION
System.Boolean	

Methods

EndDialog()

Declaration

```
public void EndDialog()
```

GenerateAndPlaySpeech(String)

Declaration

```
public IEnumerator GenerateAndPlaySpeech(string line)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	line	

Returns

TYPE	DESCRIPTION
System.Collections.IEnumerator	

GenerateReply(String, String, String)

Declaration

```
public IEnumerator GenerateReply(string otherName, string otherPersona, string line)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	otherName	
System.String	otherPersona	
System.String	line	

Returns

TYPE	DESCRIPTION
System.Collections.IEnumerator	

HandleLine(String, String, String)

Declaration

```
public void HandleLine(string otherName, string otherPersona, string line)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	otherName	
System.String	otherPersona	
System.String	line	

HandleLineCoroutine(String, String, String)

Declaration

```
public IEnumerator HandleLineCoroutine(string otherName, string otherPersona, string line)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	otherName	
System.String	otherPersona	
System.String	line	

Returns

TYPE	DESCRIPTION
System.Collections.IEnumerator	

HandlePlayerLineCoroutine(String, String, String)

Declaration

```
public IEnumerator HandlePlayerLineCoroutine(string otherName, string otherPersona, string line)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	otherName	
System.String	otherPersona	
System.String	line	

Returns

TYPE	DESCRIPTION
System.Collections.IEnumerator	

SayNPCLines()

Declaration

```
public IEnumerator SayNPCLines()
```

Returns

TYPE	DESCRIPTION
System.Collections.IEnumerator	

StartDialogue()

Declaration

```
public void StartDialogue()
```

Class NPCEngineSTT

Inheritance

System.Object

[AbstractSpeechToText](#)

NPCEngineSTT

Inherited Members

[AbstractSpeechToText.Context](#)

[AbstractSpeechToText.OnSpeechRecognized](#)

[AbstractSpeechToText.OnSpeechRecognitionFailed](#)

[AbstractSpeechToText.SpeechRecognized\(String\)](#)

Namespace: [NPCEngine.Components](#)

Assembly: cs.temp.dll.dll

Syntax

```
public class NPCEngineSTT : AbstractSpeechToText
```

Methods

CallSpeechToText()

Declaration

```
public void CallSpeechToText()
```

StartListening()

Declaration

```
public override void StartListening()
```

Overrides

[AbstractSpeechToText.StartListening\(\)](#)

StopListening()

Declaration

```
public override void StopListening()
```

Overrides

[AbstractSpeechToText.StopListening\(\)](#)

Class PlayerCharacter

Inheritance

System.Object

PlayerCharacter

Namespace: [NPCEngine.Components](#)

Assembly: cs.temp.dll.dll

Syntax

```
public class PlayerCharacter : Singleton<PlayerCharacter>
```

Fields

characterName

Declaration

```
public string characterName
```

Field Value

TYPE	DESCRIPTION
System.String	

CheckCamera

Declaration

```
public Camera CheckCamera
```

Field Value

TYPE	DESCRIPTION
Camera	

HorizontalMargin

Declaration

```
public float HorizontalMargin
```

Field Value

TYPE	DESCRIPTION
System.Single	

MaxRange

Declaration

```
public float MaxRange
```

Field Value

TYPE	DESCRIPTION
System.Single	

persona

Declaration

```
public string persona
```

Field Value

TYPE	DESCRIPTION
System.String	

settingDescription

Declaration

```
public string settingDescription
```

Field Value

TYPE	DESCRIPTION
System.String	

settingName

Declaration

```
public string settingName
```

Field Value

TYPE	DESCRIPTION
System.String	

SpeechToText

Declaration

```
public AbstractSpeechToText SpeechToText
```

Field Value

TYPE	DESCRIPTION
AbstractSpeechToText	

VerticalMargin

Declaration

```
public float VerticalMargin
```

Field Value

TYPE	DESCRIPTION
System.Single	

Methods

CheckIsSeen(Vector3)

Declaration

```
public bool CheckIsSeen(Vector3 dialogAgentPosition)
```

Parameters

TYPE	NAME	DESCRIPTION
Vector3	dialogAgentPosition	

Returns

TYPE	DESCRIPTION
System.Boolean	

DeregisterDialogueCandidate(NonPlayerCharacter)

Declaration

```
public void DeregisterDialogueCandidate(NonPlayerCharacter agent)
```

Parameters

TYPE	NAME	DESCRIPTION
NonPlayerCharacter	agent	

IsRegistered(NonPlayerCharacter)

Declaration

```
public bool IsRegistered(NonPlayerCharacter agent)
```

Parameters

TYPE	NAME	DESCRIPTION
NonPlayerCharacter	agent	

Returns

TYPE	DESCRIPTION
System.Boolean	

LeaveDialog()

Declaration

```
public void LeaveDialog()
```

RegisterDialogueCandidate(NonPlayerCharacter)

Declaration

```
public void RegisterDialogueCandidate(NonPlayerCharacter agent)
```

Parameters

TYPE	NAME	DESCRIPTION
NonPlayerCharacter	agent	

Namespace NPCEngine.Server

Classes

[NPCEngineServer](#)

Class `NPCEngineServer` manages inference engine sidecart process lifetime and communication.

[RPCRequestMessage<ParametersType>](#)

[RPCResponseError](#)

[RPCResponseMessage<ReturnType>](#)

Class NPCEngineServer

Class `NPCEngineServer` manages inference engine sidecart process lifetime and communication.

Inheritance

System.Object

NPCEngineServer

Namespace: `NPCEngine.Server`

Assembly: `cs.temp.dll.dll`

Syntax

```
public class NPCEngineServer : Singleton<NPCEngineServer>
```

Fields

connectToExistingServer

Declaration

```
public bool connectToExistingServer
```

Field Value

TYPE	DESCRIPTION
System.Boolean	

debug

Declaration

```
public bool debug
```

Field Value

TYPE	DESCRIPTION
System.Boolean	

initializeOnStart

Declaration

```
public bool initializeOnStart
```

Field Value

TYPE	DESCRIPTION
System.Boolean	

modelsPath

Declaration

```
public string modelsPath
```

Field Value

TYPE	DESCRIPTION
System.String	

npcEnginePath

Declaration

```
public string npcEnginePath
```

Field Value

TYPE	DESCRIPTION
System.String	

Properties

Initialized

Declaration

```
public bool Initialized { get; }
```

Property Value

TYPE	DESCRIPTION
System.Boolean	

Methods

ConnectToServer()

Declaration

```
public void ConnectToServer()
```

Run<P, R>(String, P)

Declaration

```
public ResultFuture<R> Run<P, R>(String methodName, P parameters)
    where P : new()
```

Parameters

TYPE	NAME	DESCRIPTION
String	methodName	
P	parameters	

Returns

TYPE	DESCRIPTION
ResultFuture<R>	

Type Parameters

NAME	DESCRIPTION
P	
R	

StartInferenceEngine()

Declaration

```
public void StartInferenceEngine()
```

Class RPCRequistMessage<ParametersType>

Inheritance

System.Object

RPCRequistMessage<ParametersType>

Namespace: [NPCEngine.Server](#)

Assembly: cs.temp.dll.dll

Syntax

```
public class RPCRequistMessage<ParametersType>
    where ParametersType : new()
```

Type Parameters

NAME	DESCRIPTION
ParametersType	

Constructors

RPCRequistMessage()

Declaration

```
public RPCRequistMessage()
```

Fields

id

Declaration

```
public int id
```

Field Value

TYPE	DESCRIPTION
System.Int32	

jsonrpc

Declaration

```
public String jsonrpc
```

Field Value

TYPE	DESCRIPTION
String	

method

Declaration

```
public String method
```

Field Value

TYPE	DESCRIPTION
String	

parameters

Declaration

public ParametersType parameters

Field Value

TYPE	DESCRIPTION
ParametersType	

Class RPCResponseError

Inheritance

System.Object

RPCResponseError

Namespace: [NPCEngine.Server](#)

Assembly: cs.temp.dll.dll

Syntax

```
public class RPCResponseError
```

Fields

code

Declaration

```
public int code
```

Field Value

TYPE	DESCRIPTION
System.Int32	

message

Declaration

```
public String message
```

Field Value

TYPE	DESCRIPTION
String	

Class RPCResponseMessage<ReturnType>

Inheritance

System.Object

RPCResponseMessage<ReturnType>

Namespace: [NPCEngine.Server](#)

Assembly: cs.temp.dll.dll

Syntax

```
public class RPCResponseMessage<ReturnType>
```

Type Parameters

NAME	DESCRIPTION
ReturnType	

Constructors

RPCResponseMessage()

Declaration

```
public RPCResponseMessage()
```

Fields

error

Declaration

```
public RPCResponseError error
```

Field Value

TYPE	DESCRIPTION
RPCResponseError	

id

Declaration

```
public int id
```

Field Value

TYPE	DESCRIPTION
System.Int32	

jsonrpc

Declaration

```
public String jsonrpc
```

Field Value

TYPE	DESCRIPTION
String	

result

Declaration

```
public ReturnType result
```

Field Value

TYPE	DESCRIPTION
ReturnType	

Namespace NPCEngine.Utility

Classes

[AudioSourceQueue](#)

[Singleton<T>](#)

Class AudioSourceQueue

Inheritance

System.Object

AudioSourceQueue

Namespace: [NPCEngine.Utility](#)

Assembly: cs.temp.dll.dll

Syntax

```
public class AudioSourceQueue : MonoBehaviour
```

Fields

audioSource

Declaration

```
public AudioSource audioSource
```

Field Value

TYPE	DESCRIPTION
AudioSource	

Methods

PlaySound(AudioClip)

Declaration

```
public void PlaySound(AudioClip clip)
```

Parameters

TYPE	NAME	DESCRIPTION
AudioClip	clip	

Class Singleton<T>

Inheritance

System.Object

Singleton<T>

Namespace: [NPCEngine.Utility](#)

Assembly: cs.temp.dll.dll

Syntax

```
public class Singleton<T> : MonoBehaviour where T : MonoBehaviour
```

Type Parameters

NAME	DESCRIPTION
T	

Properties

Instance

Declaration

```
public static T Instance { get; }
```

Property Value

TYPE	DESCRIPTION
T	