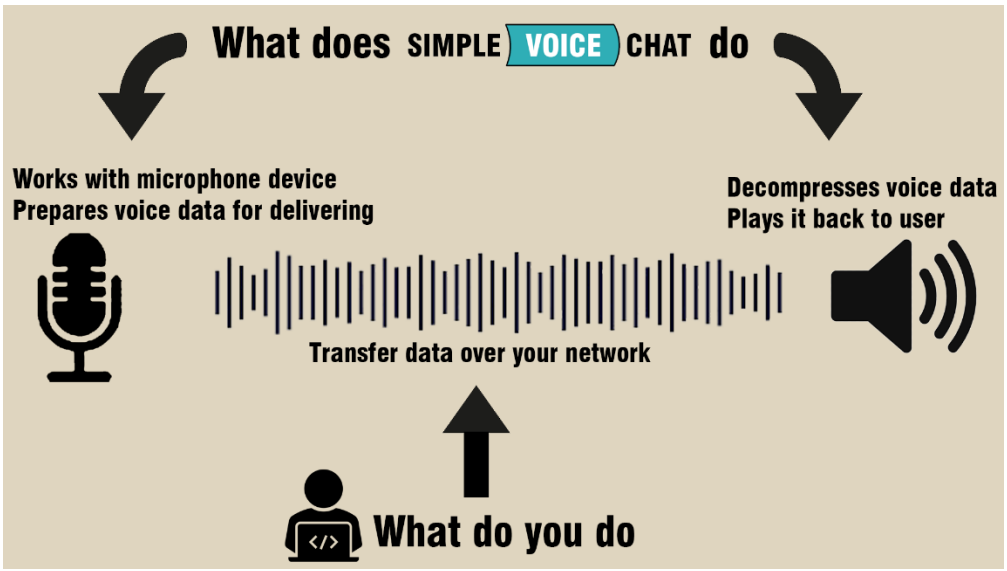
Simple and basic solution for online voice chat.

Table of content

How this package works 2
Known limitations of this package 2
Quick Start 3
Settings 4
Example scene 5

How this package works

Using this package requires you to have some networking knowledge. At least, you should be able to transfer a byte array across your network. What this package does, and what you do, is depicted in this diagram:



Known limitations of this package.

- Doesn't work with WebGL (at least until you find a way to capture audio from a microphone in WebGL).
- No echo cancelation or noise reduction (so, using with mobile speakers is not particularly comfortable)
- Requires writing code (to transfer data through your network solution)

Quick Start

- Place prefab “**Recorder**” to scene. Make sure it is active when scene starts. Use its public methods for start and stop voice transmitting.

```
public bool StartRecord() ...
public bool StopRecord() ...
public void SwitchState() ...
```

For example:

```
Recorder.Instance.StartRecord();
Recorder.Instance.StopRecord();
```

- Implement real data transmission over the network. To achieve this, subscribe to public event OnSendDataToNetwork in **Recorder** class.

```
public static event Action<byte[]> OnSendDataToNetwork;
```

This event is launched every frame, so pieces of transmitted data are small.

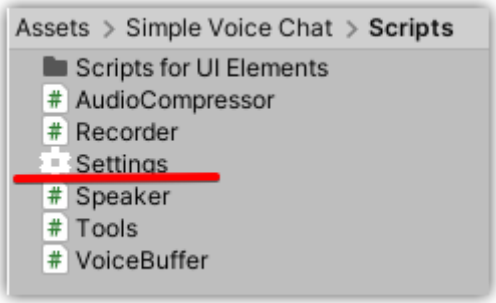
- After transmitting data over network, direct it to speaker. **Speaker** is script with AudioSource attached processing audio playback. You should have a separate speaker for each incoming audio stream from other users. For example, if your user is talking to 4 other people, then there should be 4 speakers in the scene. Speaker prefab can be found in ‘Prefabs’ folder.

Direct voice data received from network to **Speaker**’s public method:

```
public void ProcessVoiceData(byte[] voiceData) ...
```

Settings

All settings for this package are stored in a static class “Settings”. To change the values just edit the script. All options have their description.



Also, in the prefabs folder you can find a ready-to-use UI panel for microphone settings. Drag this prefab onto the scene to allow the user to select a microphone.

Example scene

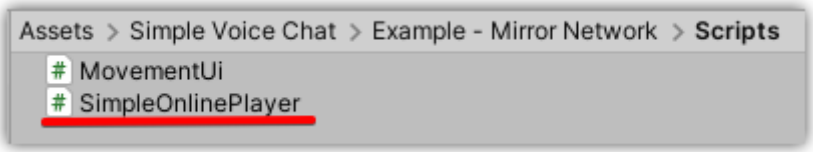
The package contains a demo scene that shows an example of using this package with the **Mirror** network package (a popular network solution for Unity). The demo scene is stored as a package. Before unpacking it, make sure that you have imported **Mirror** into your project, otherwise you will get a lot of errors in the console.

To try out the example, you need to do the following steps:

- Make a build and run it on two different devices (if you just run the application 2 times on same device, the first copy of the program will capture the microphone and the second copy will no longer be able to use it).
- Make sure both devices are connected to the same Wi-Fi network
- After launching the application on the first device, in the upper left corner, click the button "***Host (Server + Client)***"

- On the second device, enter the address of the first device and press the button "***Client***"
- You will see two characters in the scene (2 online players). Use the button at the bottom of the screen to turn on the microphone and talk. Sound must travel from one device to another through the network.

To understand how sound is transmitted through the Mirror network in this example, examine the script "**SimpleOnlinePlayer**".



You will see that when creating a player prefab for a local user, a subscription to a public event from the "Recorder" class occurs. Then data is transferred via the network using specific **Mirror**'s methods.

```
public override void OnStartLocalPlayer() {
    base.OnStartLocalPlayer();
    localPlayer = this;
    camera.gameObject.SetActive(true);

    Recorder.OnSendDataToNetwork += Cmd_SendVoiceToServer;
}
```

For more information about **Mirror** check link below:
https://assetstore.unity.com/packages/tools/network/mirror-129321

Developed by: **Oleksandr Martysh**
Contacts: ***martysh@yahoo.com***