

Endergebnis

Projekttitle: **Voice Chat Implementierung in Unity**
Auftraggeber*in: **Gerhard Sturm**
Auftragnehmer*in: **Kristian Sendele**
Schuljahr: **2025/26** Klasse: **5BI**

VERSION	DATUM	AUTORIN/AUTOR	ÄNDERUNG
v1.0	10.03.2026	Kristian Sendele	Erstellung des Dokumentes

1 Projektausgangssituation

Als neugieriger Schüler, der sich bereits teilweise mit Unity Multiplayer auseinandergesetzt hat, wollte ich mehr über das Thema Voice-Chat in Unity erfahren und auch einen selbst implementieren können.

2 Projektidee

In diesem Projekt soll ein voll funktionsfähiger Voice-Chat in Unity implementiert werden. Mindestens zwei Spieler beziehungsweise Clients sollen von unterschiedlichen Geräten, ohne physische Verbindung, flüssig miteinander kommunizieren können. Außerdem soll eine Mute-Funktion vorhanden sein, da diese für Voice-Chats üblich ist.

3 Fazit

Rückblickend war dieses Projekt eine interessante Erfahrung, weil es mir einen Einblick in die technische Umsetzung von Multiplayer-Voice-Chat gegeben hat. Als Voice-Chat-Lösung habe ich mich für Vivox entschieden. Am Anfang war die Implementierung schwieriger als erwartet. Besonders das Einrichten der Unity Gaming Services, das Verbinden mit den Voice-Channels und das Verbinden dieser Voice-Channels mit Multiplayer-Funktionen (laufen auf anderen Servern), war nicht einfach. Viele Funktionen müssen selbst implementiert werden, was am Anfang etwas Zeit benötigt, da man sich in Vivox einlesen und einarbeiten muss.

Sobald man sich jedoch in das System eingearbeitet hat, wird deutlich, dass Vivox sehr viele Möglichkeiten bietet. Der Dienst übernimmt viele komplexe technische Aufgaben wie Audioübertragung, Netzwerkkommunikation und Sprachverarbeitung, wodurch man sich stärker auf die Integration im Spiel konzentrieren kann. Gleichzeitig kann man viele Funktionen selbst gestalten, zum Beispiel Benutzeroberflächen oder Mute-Funktionen.

Die Implementierung meines Voice-Chats verlief insgesamt halbwegs gut. Nachdem die grundlegende Struktur funktionierte, war es deutlich einfacher, weitere Funktionen einzubauen und das System zu erweitern. Besonders hilfreich war, dass Vivox bereits gut in Unity integriert ist und daher gut mit anderen Unity-Systemen, wie zum Beispiel den Multiplayer-Systemen (Relay+Lobby) kombiniert werden kann. Ein wichtiger Teil des Projekts war daher nicht nur die reine Voice-Chat-Integration, sondern auch das Zusammenspiel zwischen Multiplayer-System, Spielszene und Voice-Kommunikation.

Insgesamt würde ich Vivox weiterempfehlen, besonders für Leute, die sich mit Unity-Multiplayer beschäftigen möchten. Am Anfang braucht man etwas Zeit, um hineinzukommen, aber sobald man das System verstanden hat, bietet es sehr viele Möglichkeiten und ist eine gute Lösung für Voice-Kommunikation in Spielen.