

Bài: Build ứng dụng Video Call trên nền tảng Web

Xem bài học trên website để ủng hộ Kteam: [Build ứng dụng Video Call trên nền tảng Web](#)

Mọi vấn đề về lỗi website làm ảnh hưởng đến bạn hoặc thắc mắc, mong muốn khóa học mới, nhằm hỗ trợ cải thiện Website. Các bạn vui lòng phản hồi đến Fanpage [How Kteam](#) nhé!

Xây dựng ứng dụng Video Call trên nền tảng Web

Trong bài trước, chúng ta đã cùng nhau [XÂY DỰNG ỨNG DỤNG VOICE CALL TRÊN NỀN TẢNG WEB](#).

Ở bài học này, Kteam sẽ hướng dẫn các bạn cách **Xây dựng một ứng dụng Video Call trên nền tảng Web đơn giản**.

Nội dung

Trong bài này, Kteam sẽ hướng dẫn bạn các nội dung chính như sau:

- Thêm Stringee SDK vào project
- Kết nối Stringee server
- Xử lý luồng tạo cuộc gọi đi
- Xử lý luồng nhận cuộc gọi đến
- Thêm các tính năng phụ: mute, đổi cam, đổi loa, tắt/mở camera
- Test lại luồng tạo , nhận cuộc gọi

Thư viện sử dụng

- Đăng kí account và tạo project mới
- Tham khảo Samples: [Video call integration demo](#)
- Code sample: [Web Code Sample](#)
- Xem document: [Getting started with Stringee Call API using Web SDK](#)

Hướng dẫn code

1. Lắng nghe event click từ **loginBtn** để lấy token và kết nối với **StringeeClient**

:

```
const client = new StringeeClient()
let call

client.on('connect', function () {
  console.log('connected')
})
client.on('authen', function (res) {
  console.log('authen', res)
  if (res.r === 0) {
    userId.textContent = res.userId
  }
})
loginBtn.addEventListener('click', function () {
  const token = accessToken.value
  client.connect(token)
})
```

2. Bổ sung cài đặt xử lý các event từ **StringeeCall**

:

```
function settingCallEvent(call1) {
  call1.on('addremotestream', function (stream) {
    // reset srcObject to work around minor bugs in Chrome and Edge.
    console.log('addremotestream');
    remoteVideo.srcObject = null;
    remoteVideo.srcObject = stream;
  });

  call1.on('addlocalstream', function (stream) {
    // reset srcObject to work around minor bugs in Chrome and Edge.
    console.log('addlocalstream');
    localVideo.srcObject = null;
    localVideo.srcObject = stream;
  });

  call1.on('signalingstate', function (state) {
    console.log('signalingstate ', state);
    var reason = state.reason;
    callStatus.textContent = reason;
  });

  call1.on('mediastate', function (state) {
    console.log('mediastate ', state);
  });

  call1.on('info', function (info) {
    console.log('on info:' + JSON.stringify(info));
  });
}
```

3. Lắng nghe event click của **callBtn** và thực hiện gọi hàm **makeCall** để thực hiện cuộc gọi.

Lưu ý tham số cuối truyền vào là **true** để gọi **videoCall**

:

```
callBtn.addEventListener('click', function () {
  const fromNumber = document.getElementById('from').value
  const toNumber = document.getElementById('to').value
  call = new StringeeCall(client, fromNumber, toNumber, true)
  console.log('stringee call', call)
  settingCallEvent(call)
  call.makeCall(function (res) {
    console.log('make call callback: ' + JSON.stringify(res));
  })
})
```

4. Lắng nghe event **incomingcall** của **StringeeClient** để xử lý khi có cuộc gọi đổ đến

:

```
client.on('incomingcall', function (incomingcall) {
  console.log('incomingcall', incomingcall);
  call = incomingcall;
  settingCallEvent(incomingcall);
  var answer = confirm('Incoming call from: ' + incomingcall.fromNumber + ', do you want to answer?');
  if (answer) {
    call.answer(function (res) {
      console.log('answer res', res);
    });
  } else {
    call.reject(function (res) {
      console.log('reject res', res);
    });
  }
});
```

5. Lắng nghe event click button **hangup** và **mute** để xử lý phần ngắt cuộc gọi và tắt tiếng

:

```
hangupBtn.addEventListener('click', function () {
  call.hangup(function (res) {
    console.log('hangup res', res);
  })
})

muteBtn.addEventListener('click', function () {
  console.log('clicked')
  const muted = call.muted
  console.log('muted', muted)
  call.mute(!muted)
})
```

Kết

Trong bài này, chúng ta đã cùng nhau xây dựng một ứng dụng Video Call trên nền tảng Web.

Ở bài sau, chúng ta sẽ tiếp tục tìm hiểu và thực hành cách [TÍCH HỢP PUSH NOTIFICATION VỚI IOS APP](#).

Cảm ơn các bạn đã theo dõi bài viết. Hãy để lại bình luận hoặc góp ý của bạn để phát triển bài viết tốt hơn. Đừng quên **“Luyện tập – Thử thách – Không ngại khó”**