



Guida per SDK sviluppatori di client Web

NICE DCV



NICE DCV: Guida per SDK sviluppatori di client Web

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

I marchi e l'immagine commerciale di Amazon non possono essere utilizzati in relazione a prodotti o servizi che non siano di Amazon, in una qualsiasi modalità che possa causare confusione tra i clienti o in una qualsiasi modalità che denigri o discrediti Amazon. Tutti gli altri marchi non di proprietà di Amazon sono di proprietà delle rispettive aziende, che possono o meno essere associate, collegate o sponsorizzate da Amazon.

Table of Contents

Cos'è l'SDK NICE DCV Web Client?	1
Prerequisiti	1
Funzionalità supportate	2
Supporto browser	2
Convenzione di Versioni	3
Nozioni di base	4
Connect a un server NICE DCV e ottieni il primo frame	5
Fase 1: Preparare la pagina HTML	5
Fase 2: Autenticazione, connessione e acquisizione del primo frame	6
Bonus: crea automaticamente un modulo di accesso HTML	9
Funziona con le funzionalità di NICE DCV	10
Comprendere la funzione di callback FeaturesUpdate	10
Gestione degli aggiornamenti delle funzionalità	11
Usa l'SDK dell'interfaccia utente Web NICE DCV	11
Prerequisiti	12
Passaggio 1: prepara la tua pagina HTML	13
Passaggio 2: autenticare, connettere e renderizzare il componente DCVViewer React.	13
Aggiornamento da AWS-UI a Cloudscape Design System	17
Riferimento SDK	19
Modulo DCV	19
Metodi	19
Membri	22
Definizioni di tipi e callback	27
Classe di connessione	67
Metodi	19
Classe di autenticazione	95
Metodi	19
Categoria risorsa	96
Metodi	19
SDK dell'interfaccia utente Web NICE DCV	97
Componenti	98
Note di rilascio e cronologia dei documenti	106
Note di rilascio	106
1.5.10 — 19 dicembre 2023	107

1.5.6 — 9 novembre 2023	107
1.4.4 — 29 giugno 2023	107
1.4.0 — 28 marzo 2023	108
1.3.1 — 9 dicembre 2022	110
1.3.0 — 11 novembre 2022	110
1.2.1 — 21 luglio 2022	111
1.2.0 — 29 giugno 2022	112
1.1.3 — 23 maggio 2022	112
1.1.2 — 19 maggio 2022	113
1.1.1 — 23 marzo 2022	113
1.1.0 — 23 febbraio 2022	113
1.0.4 — 20 dicembre 2021	114
1.0.3 — 01 settembre 2021	115
1.0.2 — 30 luglio 2021	116
1.0.1 — 31 maggio 2021	116
1.0.0 — 24 marzo 2021	116
Cronologia dei documenti	117
.....	cxix

Cos'è l'SDK NICE DCV Web Client?

NICE DCV è un protocollo di visualizzazione remota ad alte prestazioni. Consente di distribuire in modo sicuro desktop remoti e streaming di applicazioni da qualsiasi cloud o data center a qualsiasi dispositivo, in condizioni di rete variabili. Utilizzando NICE DCV con Amazon EC2, puoi eseguire applicazioni ad alta intensità di grafica in remoto su istanze Amazon EC2. È quindi possibile eseguire lo streaming dei risultati su macchine client più modeste, eliminando così la necessità di costose workstation dedicate.

Il NICE DCV Web Client SDK è una JavaScript libreria che puoi utilizzare per sviluppare le tue applicazioni client per browser Web NICE DCV. Gli utenti finali possono utilizzare queste applicazioni per connettersi e interagire con una sessione NICE DCV in esecuzione.

Utilizzando il NICE DCV Web Client SDK come elemento costitutivo, è possibile creare applicazioni Web personalizzate che forniscono agli utenti l'accesso immediato al desktop o alle applicazioni da qualsiasi luogo, con prestazioni reattive e fluide quasi indistinguibili da un'applicazione installata in modo nativo.

Questa guida spiega come utilizzare l'SDK NICE DCV Web Client per creare applicazioni client personalizzate per browser Web per interagire con le sessioni NICE DCV all'interno dei flussi di lavoro.

Argomenti

- [Prerequisiti](#)
- [Funzionalità supportate](#)
- [Supporto browser](#)
- [Convenzione di Versioni](#)

Prerequisiti

Prima di iniziare a lavorare con l'SDK NICE DCV Web Client, assicurati di avere familiarità con le sessioni NICE DCV e NICE DCV. Per ulteriori informazioni, consulta la [Guida per l'amministratore di NICE DCV](#).

Il NICE DCV Web Client SDK supporta il server NICE DCV versione 2020 e successive.

Funzionalità supportate

È possibile creare applicazioni client personalizzate per browser Web che supportano le seguenti funzionalità NICE DCV:

- Connect ai server Windows NICE DCV
- Connect a server Linux NICE DCV
- Gestione delle modalità di streaming
- Trasferimento dei file
- Stampa da sessioni
- Copia e incolla
- Riproduzione audio Stereo 2.0
- Registrazione audio Stereo 2.0 (su server Windows)
- Touchscreen
- Stylus (su server Linux, Windows 10 e Windows Server 2019)
- Supporto per più monitor

Per ulteriori informazioni su queste funzionalità, vedere Funzioni [supportate](#) nella Guida per l'utente di NICE DCV.

Supporto browser

Il NICE DCV Web Client SDK supporta JavaScript (ES6) e può essere utilizzato dalle JavaScript nostreTypeScript applicazioni.

L'SDK NICE DCV Web Client supporta i seguenti browser Web:

Browser	Versione
Google Chrome	Versioni principali di.
Mozilla Firefox	Versioni principali di.
Microsoft Edge	Versioni principali di.

Browser	Versione
Apple Safari per macOS	Versioni principali di.

Convenzione di Versioni

La versione NICE DCV Web Client SDK è definita nel seguente formato: *major.minor.patch*. La convenzione di controllo delle versioni generalmente aderisce al [modello di versionamento semantico](#). Una modifica nella versione principale, ad esempio da 1.x.x a 2.x.x, indica che sono state introdotte modifiche sostanziali che potrebbero richiedere modifiche al codice e una distribuzione pianificata. Una modifica nella versione secondaria, ad esempio da 1.1.x a 1.2.x, è compatibile con le versioni precedenti, ma potrebbe includere elementi obsoleti.

Guida introduttiva al NICE DCV Web Client SDK

L'SDK NICE DCV Web Client comprende un `dcv.js` file principale e alcuni componenti ausiliari.

[Tutti i file sono distribuiti all'interno di un archivio compresso che può essere scaricato dal sito Web di NICE.](#)

Per iniziare a usare il NICE DCV Web Client SDK

1. L'archivio NICE DCV Web Client SDK è firmato digitalmente con una firma GPG sicura. Per verificare la firma dell'archivio, è necessario importare la chiave NICE GPG. Per farlo, aprite una finestra di terminale e importate la chiave NICE GPG.

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/NICE-GPG-KEY
```

```
$ gpg --import NICE-GPG-KEY
```

2. [Scarica l'archivio NICE DCV Web Client SDK e la firma di archivio NICE DCV Web Client SDK dal sito Web NICE.](#)
3. Verifica la firma dell'archivio NICE DCV Web Client SDK utilizzando la firma.

```
$ gpg --verify  
signature_filename.zip.sign  
archive_filename.zip
```

Per esempio:

```
$ gpg --verify nice-dcv-web-client-sdk-1.5.10-684.zip.sign nice-dcv-web-client-  
sdk-1.5.10-684.zip
```

4. Se la firma viene verificata con successo, estrai il contenuto dell'archivio NICE DCV Web Client SDK e posiziona la directory estratta sul tuo server web. Per esempio:

```
$ unzip  
archive_filename.zip  
-d /  
path_to
```



```
/
  server_directory
/
```

Important

- È necessario mantenere la struttura delle cartelle quando si distribuisce l'SDK NICE DCV Web Client sul server Web.
- Quando usi NICE DCV Web UI SDK, fai attenzione che il componente `DCVViewer` React si aspetta che i file `EULA.txt` `third-party-licenses` e `e.txt` di questo pacchetto siano presenti nel percorso URL del server web incorporato. Il `third-party-licenses` file.txt deve essere modificato per includere anche il contenuto del file corrispondente del pacchetto NICE DCV Web Client SDK e possibilmente qualsiasi altra informazione sulla licenza dalle librerie utilizzate dall'applicazione utente che consuma.

Connect a un server NICE DCV e ottieni il primo frame

Il seguente tutorial mostra come preparare la pagina HTML per il client web personalizzato, come autenticarsi e connettersi a un server NICE DCV e come ricevere il primo frame di contenuto in streaming dalla sessione NICE DCV.

Argomenti

- [Fase 1: Preparare la pagina HTML](#)
- [Fase 2: Autenticazione, connessione e acquisizione del primo frame](#)
- [Bonus: crea automaticamente un modulo di accesso HTML](#)

Fase 1: Preparare la pagina HTML

Nella pagina Web, è necessario caricare i JavaScript moduli necessari e aggiungere un elemento `<div>` HTML con un valore valido `id` in cui si desidera che l'SDK NICE DCV Web Client attiri il flusso di contenuti dal server NICE DCV remoto.

Per esempio:

```
<!DOCTYPE html>
```

```
<html lang="en" style="height: 100%;">
  <head>
    <title>DCV first connection</title>
  </head>
  <body style="height: 100%;">
    <div id="root" style="height: 100%;"></div>
    <div id="dcv-display"></div>
    <script type="module" src="index.js"></script>
  </body>
</html>
```

Fase 2: Autenticazione, connessione e acquisizione del primo frame

Questa sezione mostra come completare il processo di autenticazione dell'utente, come connettere il server NICE DCV e come ricevere il primo frame di contenuto dal server NICE DCV.

Innanzitutto, dal `index.js` file importate il NICE DCV Web Client SDK. Può essere importato sia come modulo Universal Module Definition (UMD), in questo modo:

```
import "../dcvjs/dcv.js"
```

Altrimenti, a partire dalla versione `1.1.0`, può anche essere importato come modulo ECMAScript (ESM) dal pacchetto corrispondente, in questo modo:

```
import dcv from "../dcvjs/dcv.js"
```

Definire le variabili da utilizzare per memorizzare l'oggetto `Authentication`, l'oggetto `Connection` e l'URL del server NICE DCV.

```
let auth,
    connection,
    serverUrl;
```

Al caricamento dello script, registra la versione di NICE DCV Web Client SDK e, al caricamento della pagina, chiama la funzione `main`

```
console.log("Using NICE DCV Web Client SDK version " + dcv.version.versionStr);
document.addEventListener('DOMContentLoaded', main);
```

La `main` funzione imposta il livello di registro e avvia il processo di autenticazione.

```
function main () {
  console.log("Setting log level to INFO");
  dcv.setLogLevel(dcv.LogLevel.INFO);

  serverUrl = "https://your-dcv-server-url:port/";

  console.log("Starting authentication with", serverUrl);

  auth = dcv.authenticate(
    serverUrl,
    {
      promptCredentials: onPromptCredentials,
      error: onError,
      success: onSuccess
    }
  );
}
```

Le funzioni `promptCredentials`, `error`, e `success` sono funzioni di callback obbligatorie che devono essere definite nel processo di autenticazione.

Se il server NICE DCV richiede le credenziali, la funzione di `promptCredentials` callback riceve la richiesta di credenziali dal server NICE DCV. Se il server NICE DCV è configurato per utilizzare l'autenticazione del sistema, è necessario fornire le credenziali di accesso. I seguenti esempi di codice presuppongono che il nome utente sia `my_dcv_user` e che la password sia `my_password`.

Se l'autenticazione fallisce, la funzione di `error` callback riceve un oggetto di errore dal server NICE DCV.

Se l'autenticazione ha esito positivo, la funzione di `success` callback riceve una serie di coppie che include l'id di sessione (`sessionId`) e i token di autorizzazione (`authToken`) per ogni sessione a cui l'`my_dcv_user` utente può connettersi sul server NICE DCV. Il seguente esempio di codice richiama la funzione `connect` e si connette alla prima sessione restituita nell'array.

Note

Nel seguente esempio di codice, `MY_DCV_USER` sostituitelo con il vostro nome utente e `MY_PASSWORD` con la vostra password.

```
function onPromptCredentials(auth, challenge) {
  // Let's check if in challenge we have a username and password request
  if (challengeHasField(challenge, "username") && challengeHasField(challenge,
"password")) {
    auth.sendCredentials({username: MY_DCV_USER, password: MY_PASSWORD})
  } else {
    // Challenge is requesting something else...
  }
}

function challengeHasField(challenge, field) {
  return challenge.requiredCredentials.some(credential => credential.name === field);
}

function onError(auth, error) {
  console.log("Error during the authentication: " + error.message);
}

// We connect to the first session returned
function onSuccess(auth, result) {
  let {sessionId, authToken} = {...result[0]};

  connect(sessionId, authToken);
}
```

Connect al server NICE DCV. Il metodo `firstFrame` callback viene chiamato quando il primo frame viene ricevuto dal server NICE DCV.

```
function connect (sessionId, authToken) {
  console.log(sessionId, authToken);

  dcv.connect({
    url: serverUrl,
    sessionId: sessionId,
    authToken: authToken,
    divId: "dcv-display",
    callbacks: {
      firstFrame: () => console.log("First frame received")
    }
  }).then(function (conn) {
    console.log("Connection established!");
    connection= conn;
  }).catch(function (error) {
```

```
    console.log("Connection failed with error " + error.message);
  });
}
```

Bonus: crea automaticamente un modulo di accesso HTML

L'challengeoggetto viene restituito quando viene `promptCredentials` chiamata la funzione di `callback`. Include una proprietà denominata `requiredCredentials` che è una matrice di oggetti: un oggetto per credenziale richiesta dal server NICE DCV. Ogni oggetto include il nome e il tipo della credenziale richiesta. È possibile utilizzare gli `requiredCredentials` oggetti `challenge` e per creare automaticamente un modulo di accesso HTML.

Il seguente esempio di codice mostra come eseguire questa operazione.

```
let form,
    fieldSet;

function submitCredentials (e) {
  var credentials = {};
  fieldSet.childNodes.forEach(input => credentials[input.id] = input.value);
  auth.sendCredentials(credentials);
  e.preventDefault();
}

function createLoginForm () {
  var submitButton = document.createElement("button");

  submitButton.type = "submit";
  submitButton.textContent = "Login";

  form = document.createElement("form");
  fieldSet = document.createElement("fieldset");

  form.onsubmit = submitCredentials;
  form.appendChild(fieldSet);
  form.appendChild(submitButton);

  document.body.appendChild(form);
}

function addInput (name) {
  var type = name === "password" ? "password" : "text";
```

```
var inputField = document.createElement("input");
inputField.name = name;
inputField.id = name;
inputField.placeholder = name;
inputField.type = type;
fieldSet.appendChild(inputField);
}

function onPromptCredentials (_, credentialsChallenge) {
  createLoginForm();
  credentialsChallenge.requiredCredentials.forEach(challenge =>
    addInput(challenge.name));
}
```

Funziona con le funzionalità di NICE DCV

La disponibilità delle funzionalità di NICE DCV dipende dalle autorizzazioni configurate per la sessione NICE DCV e dalle funzionalità del browser Web del client.

Le funzionalità disponibili in una sessione NICE DCV sono gestite dalle autorizzazioni specificate per la sessione. Ciò significa che anche se una funzionalità è supportata dall'SDK NICE DCV Web Client, l'accesso a tale funzionalità potrebbe essere impedito in base alle autorizzazioni definite dall'amministratore della sessione. Per ulteriori informazioni, vedere [Configurazione dell'autorizzazione NICE DCV](#) nella Guida per l'amministratore di NICE DCV.

Comprendere la funzione di callback FeaturesUpdate

Quando la disponibilità di una funzionalità in una sessione NICE DCV cambia, l'SDK NICE DCV Web Client ti avvisa utilizzando `featuresUpdate` la funzione di callback specificata al momento di stabilire la connessione. Per esempio:

```
featuresUpdate: function (connection, list) {
  ...
},
```

La funzione di callback ti notifica solo le funzionalità per le quali la disponibilità è cambiata. Il `list` parametro è una matrice di stringhe e include solo i nomi delle funzionalità aggiornate. Ad esempio, se la disponibilità della funzione di ingresso audio cambia per la sessione, il parametro include

solo["audio-in"]. Se in un secondo momento, la disponibilità delle funzioni di copia e incolla degli appunti cambia per la sessione, il parametro include solo. ["clipboard-copy", "clipboard-paste"]

Gestione degli aggiornamenti delle funzionalità

La funzione di `featuresUpdate` callback avvisa solo che la disponibilità di una o più funzionalità è cambiata. Per sapere quali funzionalità sono state aggiornate, è necessario interrogare la funzionalità utilizzando il `connection.queryFeature` metodo. Questa operazione può essere eseguita in qualsiasi momento dopo la ricezione della notifica di modifica. Questo metodo restituisce un `Promise` messaggio che si risolve nello stato aggiornato della funzionalità richiesta. Il `status` valore è sempre associato e ha una proprietà booleana (`true|false`) chiamata `enabled`. Alcune funzionalità potrebbero avere proprietà aggiuntive nel `status` valore. Se la disponibilità della funzionalità non è stata aggiornata, viene rifiutata.

Il codice di esempio seguente mostra come eseguire questa operazione.

```
// Connection callback called
function featuresUpdate (_, list) {
  if (list.length > 0) {
    list.forEach((feat) => {
      connection.queryFeature(feat).then(status => console.log(feat, "is",
        status.enabled));
    });
  }
}
```

Usa l'SDK dell'interfaccia utente Web NICE DCV

Il seguente tutorial mostra come autenticarsi sul server NICE DCV, connettersi ad esso ed eseguire il rendering del componente React dall'`DCVViewerSDK` dell'interfaccia utente Web NICE DCV.

Argomenti

- [Prerequisiti](#)
- [Passaggio 1: prepara la tua pagina HTML](#)
- [Passaggio 2: autenticare, connettere e renderizzare il componente DCVViewer React.](#)
- [Aggiornamento da AWS-UI a Cloudscape Design System](#)

Prerequisiti

È necessario installare React, eReactDOM. Cloudscape Design Components React Cloudscape Design Global Styles Cloudscape Design Design Tokens

```
$ npm i react react-dom @cloudscape-design/components @cloudscape-design/global-styles @cloudscape-design/design-tokens
```

Dovresti anche scaricare NICE DCV Web Client SDK. Leggi [Guida introduttiva al NICE DCV Web Client SDK](#) la step-by-step guida su come farlo.

È necessario creare un alias per importare il dcv modulo, poiché si tratta di una dipendenza esterna per NICE DCV Web UI SDK. [Ad esempio, se stai usando webpack per raggruppare la tua app web, puoi usare l'opzione resolve.alias in questo modo:](#)

```
const path = require('path');

module.exports = {
  //...
  resolve: {
    alias: {
      dcv: path.resolve('path', 'to', 'dcv.js'),
    },
  },
};
```

[Se stai usando rollup per il raggruppamento, puoi installare @rollup /plugin-alias e usarlo in questo modo:](#)

```
import alias from '@rollup/plugin-alias';
const path = require('path');

module.exports = [
  //...
  plugins: [
    alias({
      entries: [
        { find: 'dcv', replacement: path.resolve('path', 'to', 'dcv.js') },
      ]
    })
  ]
];
```



```
};
```

Passaggio 1: prepara la tua pagina HTML

Nella tua pagina web, devi caricare i JavaScript moduli richiesti e dovresti avere un elemento `<div>` HTML con un valore valido `id` in cui verrà visualizzato il componente di ingresso della tua app.

Per esempio:

```
<!DOCTYPE html>
<html lang="en" style="height: 100%;">
  <head>
    <title>DCV first connection</title>
  </head>
  <body style="height: 100%;">
    <div id="root" style="height: 100%;"></div>
    <script type="module" src="index.js"></script>
  </body>
</html>
```

Passaggio 2: autenticare, connettere e renderizzare il componente **DCVViewer** React.

Questa sezione mostra come completare il processo di autenticazione dell'utente, come connettere il server NICE DCV e come renderizzare il componente `DCVViewer` React.

Innanzitutto, dal `index.js` file, dall'importazione `React` `ReactDOM` e dal `App` componente di primo livello.

```
import React from "react";
import ReactDOM from 'react-dom';
import App from './App';
```

Esegui il rendering del nodo contenitore di primo livello della tua app.

```
ReactDOM.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>,
  document.getElementById("root")
```

```
);
```

Nel `App.js` file, importa l'SDK NICE DCV Web Client come modulo ESM, il componente `DCVViewer` React dall'SDK dell'interfaccia utente Web NICE DCV e il pacchetto. `React Cloudscape Design Global Styles`

```
import React from "react";
import dcv from "dcv";
import "@cloudscape-design/global-styles/index.css";
import {DCVViewer} from "../dcv-ui/dcv-ui.js";
```

Di seguito è riportato un esempio che mostra come autenticarsi con il NICE DCV Server e renderizzare `DCVViewer` il componente React da NICE DCV Web UI SDK, a condizione che l'autenticazione abbia avuto successo.

```
const LOG_LEVEL = dcv.LogLevel.INFO;
const SERVER_URL = "https://your-dcv-server-url:port/";
const BASE_URL = "/static/js/dcvjs";

let auth;

function App() {
  const [authenticated, setAuthenticated] = React.useState(false);
  const [sessionId, setSessionId] = React.useState('');
  const [authToken, setAuthToken] = React.useState('');
  const [credentials, setCredentials] = React.useState({});

  const onSuccess = (_, result) => {
    var { sessionId, authToken } = { ...result[0] };

    console.log("Authentication successful.");

    setSessionId(sessionId);
    setAuthToken(authToken);
    setAuthenticated(true);
    setCredentials({});
  }

  const onPromptCredentials = (_, credentialsChallenge) => {
    let requestedCredentials = {};
  }
}
```

```
credentialsChallenge.requiredCredentials.forEach(challenge =>
requestedCredentials[challenge.name] = "");
setCredentials(requestedCredentials);
}

const authenticate = () => {
  dcv.setLogLevel(LOG_LEVEL);

  auth = dcv.authenticate(
    SERVER_URL,
    {
      promptCredentials: onPromptCredentials,
      error: onError,
      success: onSuccess
    }
  );
}

const updateCredentials = (e) => {
  const { name, value } = e.target;
  setCredentials({
    ...credentials,
    [name]: value
  });
}

const submitCredentials = (e) => {
  auth.sendCredentials(credentials);
  e.preventDefault();
}

React.useEffect(() => {
  if (!authenticated) {
    authenticate();
  }
}, [authenticated]);

const handleDisconnect = (reason) => {
  console.log("Disconnected: " + reason.message + " (code: " + reason.code + ")");
  auth.retry();
  setAuthenticated(false);
}

return (
```

```
authenticated ?
<DCVViewer
  dcv={{
    sessionId: sessionId,
    authToken: authToken,
    serverUrl: SERVER_URL,
    baseUrl: BASE_URL,
    onDisconnect: handleDisconnect,
    logLevel: LOG_LEVEL
  }}
  uiConfig={{
    toolbar: {
      visible: true,
      fullscreenButton: true,
      multimonitorButton: true,
    },
  }}
/>
:
<div
  style={{
    height: window.innerHeight,
    backgroundColor: "#373737",
    display: 'flex',
    alignItems: 'center',
    justifyContent: 'center',
  }}
>
  <form>
    <fieldset>
      {Object.keys(credentials).map((cred) => (
        <input
          key={cred}
          name={cred}
          placeholder={cred}
          type={cred === "password" ? "password" : "text"}
          onChange={updateCredentials}
          value={credentials[cred]}
        />
      ))}
    </fieldset>
    <button
      type="submit"
      onClick={submitCredentials}
```

```
        >
          Login
        </button>
      </form>
    </div>
  );
}

const onError = (_, error) => {
  console.log("Error during the authentication: " + error.message);
}

export default App;
```

Le funzioni `promptCredentialError`, e sono `success` funzioni di callback obbligatorie che devono essere definite nel processo di autenticazione.

Se il server NICE DCV richiede le credenziali, la funzione di `promptCredentials` callback riceve la richiesta di credenziali dal server NICE DCV. Se il server NICE DCV è configurato per utilizzare l'autenticazione del sistema, le credenziali devono essere fornite sotto forma di nome utente e password.

Se l'autenticazione fallisce, la funzione di `error` callback riceve un oggetto di errore dal server NICE DCV.

Se l'autenticazione ha esito positivo, la funzione di `success` callback riceve una serie di coppie che include l'id di sessione (`sessionId`) e i token di autorizzazione (`authToken`) per ogni sessione a cui l'utente può connettersi sul server NICE DCV. L'esempio di codice riportato sopra aggiorna lo stato di React per rendere il `DCVViewer` componente in caso di autenticazione riuscita.

Per saperne di più sulle proprietà accettate da questo componente, consultate il riferimento all'[SDK dell'interfaccia utente Web NICE DCV](#).

Per ulteriori informazioni sui certificati autofirmati, consulta i [chiarimenti sul reindirizzamento](#) con certificati autofirmati.

Aggiornamento da AWS-UI a Cloudscape Design System

[A partire dalla versione SDK 1.3.0 abbiamo aggiornato il nostro `DCVViewer` componente da AWS-UI alla sua evoluzione: Cloudscape Design.](#)

Cloudscape utilizza un tema visivo diverso rispetto a AWS-UI, ma la base di codice sottostante rimane la stessa. Pertanto, la migrazione dell'applicazione basata su dovrebbe essere semplice. DCVViewer Per migrare, sostituisci i pacchetti NPM relativi all'AWS-UI che hai installato con i pacchetti Cloudscape associati:

Nome del pacchetto AWS-UI	Nome del pacchetto Cloudscape
@awsui /components-react	@cloudscape -design/componenti
@awsui /global-styles	@cloudscape -design/global-styles
@awsui /collection-hooks	@cloudscape -design/collezione-ganci
@awsui /design-tokens	@cloudscape -design/design-tokens

Per ulteriori dettagli sulla migrazione, consulta la pagina della documentazione di [AWS-UI GitHub](#) .

Riferimento SDK

Questa sezione fornisce descrizioni, sintassi ed esempi di utilizzo per il NICE DCV Web Client SDK.

Argomenti

- [Modulo DCV](#)
- [Classe di connessione](#)
- [Classe di autenticazione](#)
- [Categoria risorsa](#)
- [SDK dell'interfaccia utente Web NICE DCV](#)

Modulo DCV

Un modulo che implementa il lato client del protocollo DCV.

Espone

- [Metodi](#)
- [Membri](#)
- [Definizioni di tipi e callback](#)

Metodi

Elenco

- [authenticate \(url, callback\) → {Autenticazione}](#)
- [connect \(configurazione\) → {Promessa. < Connessione >|Promessa. < {code: ConnectionErrorCode, messaggio: string} >}](#)
- [setLogHandler\(gestore\) → {void}](#)
- [setLogLevel\(livello\) → {void}](#)

`authenticate (url, callback) → {Autenticazione}`

Avvia il processo di autenticazione per l'endpoint del server NICE DCV specificato.

Parametri:

Nome	Type (Tipo)	Descrizione
url	string	Il nome host e la porta del server NICE DCV in esecuzione nel seguente formato: <code>https://dcv_host_address:port</code> . Ad esempio: <code>https://my-dcv-server:8443</code> .
callbacks	Richiamate di autenticazione	I callback disponibili per essere chiamati durante il processo di autenticazione.

Valori restituiti:

- L'oggetto di autenticazione.

Type (Tipo)

[Autenticazione](#)

`connect (configurazione) → {Promessa. < Connessione > | Promessa. < {code: ConnectionErrorCode, messaggio: string} >}`

Si connette all'endpoint del server NICE DCV specificato. Se la connessione viene eseguita correttamente, restituisce un oggetto `Connection`. Se la connessione fallisce, restituisce un oggetto di errore.

Parametri:

Nome	Type (Tipo)	Descrizione
config	ConnectionConfig	L' <code>ConnectionConfig</code> oggetto.

Valori restituiti:

- Un oggetto Connection o un oggetto di errore.

Type (Tipo)

Promessa. < [Connessione](#) > | Promessa. < {code: [ConnectionErrorCode](#), messaggio: stringa} >

setLogHandler(gestore) → {void}

Imposta una funzione di gestione dei log personalizzata. Quando si sovrascrive il gestore di registro predefinito, la posizione di immissione del registro originale verrà persa durante il debug con la console del browser.

Parametri:

Nome	Type (Tipo)	Descrizione
handler	funzione	La funzione di gestione dei log personalizzata. La funzione handler contiene level (numero), levelName (string), domain (string) e message (string).

Valori restituiti:**Type (Tipo)**

void

setLogLevel(livello) → {void}

Imposta il livello di registro. Ciò è necessario solo se viene utilizzato il gestore di log predefinito.

Parametri:

Nome	Type (Tipo)	Descrizione
level	LogLevel	Il livello di log da usare.

Valori restituiti:

Type (Tipo)

void

Membri

Elenco

- [\(costante\) AudioError: AudioErrorCode](#)
- [\(costante\) AuthenticationError: AuthenticationErrorCode](#)
- [\(costante\) ChannelError: ChannelErrorCode](#)
- [\(costante\) ClosingReasonError: ClosingReasonErrorCode](#)
- [\(costante\) ConnectionError: ConnectionErrorCode](#)
- [\(costante\) CustomChannelError: CustomChannelErrorCode](#)
- [\(costante\) DisplayConfigError: DisplayConfigErrorCode](#)
- [\(costante\) FileStorageError: FileStorageErrorCode](#)
- [\(costante\) LogLevel: LogLevel](#)
- [\(costante\) MultiMonitorError: MultiMonitorErrorCode](#)
- [\(costante\) ResolutionError: ResolutionErrorCode](#)
- [\(costante\) TimezoneRedirectionError: TimezoneRedirectionErrorCode](#)
- [\(costante\) TimezoneRedirectionSetting: TimezoneRedirectionSettingCode](#)
- [\(costante\) TimezoneRedirectionStatus: TimezoneRedirectionStatusCode](#)
- [versione \(costante\)](#)
- [\(costante\) ScreenshotError: ScreenshotErrorCode](#)
- [\(costante\) WebcamError: WebcamErrorCode](#)

(costante) AudioError: [AudioErrorCode](#)

I AudioError codici enum.

Type:

- [AudioErrorCode](#)

(costante) AuthenticationError: [AuthenticationErrorCode](#)

I AuthenticationError codici enum.

Type:

- [AuthenticationErrorCode](#)

(costante) ChannelError: [ChannelErrorCode](#)

I ChannelError codici enum.

Type:

- [ChannelErrorCode](#)

(costante) ClosingReasonError: [ClosingReasonErrorCode](#)

I ClosingReasonError codici enum.

Type:

- [ClosingReasonErrorCode](#)

(costante) ConnectionError: [ConnectionErrorCode](#)

I ConnectionError codici enum.

Type:

- [ConnectionErrorCode](#)

(costante) CustomChannelError: [CustomChannelErrorCode](#)

I CustomChannelError codici enum.

Type:

- [CustomChannelErrorCode](#)

(costante) DisplayConfigError: [DisplayConfigErrorCode](#)

I DisplayConfigError codici enum.

Type:

- [DisplayConfigErrorCode](#)

(costante) FileStorageError: [FileStorageErrorCode](#)

I FileStorageError codici enum.

Type:

- [FileStorageErrorCode](#)

(costante) LogLevel: [LogLevel](#)

I livelli di registro SDK disponibili.

Type:

- [LogLevel](#)

(costante) MultiMonitorError: [MultiMonitorErrorCode](#)

I MultiMonitorError codici enum.

Type:

- [MultiMonitorErrorCode](#)

(costante) ResolutionError: [ResolutionErrorCode](#)

I ResolutionError codici enum.

Type:

- [ResolutionErrorCode](#)

(costante) TimezoneRedirectionError: [TimezoneRedirectionErrorCode](#)

I TimezoneRedirectionError codici enum.

Type:

- [TimezoneRedirectionErrorCode](#)

(costante) TimezoneRedirectionSetting: [TimezoneRedirectionSettingCode](#)

I TimezoneRedirectionSetting codici enum.

Type:

- [TimezoneRedirectionSettingCode](#)

(costante) TimezoneRedirectionStatus: [TimezoneRedirectionStatusCode](#)

I TimezoneRedirectionStatus codici enum.

Type:

- [TimezoneRedirectionStatusCode](#)

versione (costante)

La versione NICE DCV con versione principale, secondaria, patch, revisione, estesa e versionSTR.

Proprietà:

Nome	Type (Tipo)	Descrizione
major	integer	Il numero di versione principale.
minor	integer	Il numero di versione minore.
patch	integer	Il numero di versione patch.
revision	integer	Il numero di revisione.
extended	string	La stringa estesa.
versionStr	string	Una concatenazione dei numeri principali, secondari, di patch e di revisione nel modulomajor.minor.patch+build.revision.

(costante) ScreenshotError: [ScreenshotErrorCode](#)

I ScreenshotError codici enum.

Type:

- [ScreenshotErrorCode](#)

(costante) WebcamError: [WebcamErrorCode](#)

I WebcamError codici enum.

Type:

- [WebcamErrorCode](#)

Definizioni di tipi e callback

Elenco

- [AudioErrorCode](#)
- [Richiamate di autenticazione](#)
- [AuthenticationErrorCode](#)
- [authErrorCallback\(autenticazione, errore\)](#)
- [authPromptCredentialsRichiamata \(autenticazione, sfida\)](#)
- [authSuccessCallback\(autenticazione, dati di autenticazione\)](#)
- [Canale](#)
- [ChannelErrorCode](#)
- [clipboardEventCallback\(evento\)](#)
- [ClosingReasonErrorCode](#)
- [Spazio cromatico](#)
- [Richiamate di connessione](#)
- [ConnectionConfig](#)
- [ConnectionErrorCode](#)
- [CreateDirectory \(percorso\)](#)
- [CustomChannelErrorCode](#)
- [dataChannelCallback\(informazioni\)](#)
- [deleteFile \(percorso\)](#)
- [deviceChangeEventRichiamata \(\)](#)
- [DisconnectCallback \(motivo\)](#)
- [displayAvailabilityCallback\(stato, DisplayID\)](#)
- [DisplayConfigErrorCode](#)
- [displayLayoutCallback\(larghezza del server, altezza del server, testate\)](#)
- [caratteristica](#)
- [featuresUpdateCallback\(Elenco delle caratteristiche\)](#)
- [fileDownloadCallback\(Risorsa di file\)](#)
- [filePrintedCallback\(Risorsa di stampa\)](#)

- [archiviazione di file](#)
- [filestorageEnabledCallback\(abilitato\)](#)
- [FileStorageErrorCode](#)
- [firstFrameCallback\(Ridimensiona relativeMouseMode abilitato, DisplayID\)](#)
- [idleWarningNotificationRichiamata \(disconnectionDateTime\)](#)
- [collaboratorListCallback\(collaboratori\)](#)
- [licenseNotificationCallback\(notifica\)](#)
- [elenco \(percorso\)](#)
- [LogLevel](#)
- [Monitor \(Monitoraggio\)](#)
- [MultiMonitorErrorCode](#)
- [qualityIndicatorStateCallback \(stato\)](#)
- [Rinomina cartella \(src, dest\)](#)
- [Rinomina file \(src, dest\)](#)
- [ResolutionErrorCode](#)
- [RetrieveFile \(percorso\)](#)
- [ScreenshotCallback \(immagine dello schermo\)](#)
- [ScreenshotErrorCode](#)
- [Informazioni sul server](#)
- [stats](#)
- [StoreFile \(file, directory\)](#)
- [TimezoneRedirectionErrorCode](#)
- [TimezoneRedirectionSettingCode](#)
- [TimezoneRedirectionStatusCode](#)
- [WebcamErrorCode](#)

AudioErrorCode

Le AudioError enumerazioni di codice disponibili nel modulo DCV

- SETTING_AUDIO_FAILED
- CHANNEL_NOT_AVAILABLE

Type:

- numero

Richiamate di autenticazione

Richiamate di autenticazione

Type:

- Oggetto

Proprietà:

Nome	Type (Tipo)	Descrizione
promptCredentials	authPromptCredentialsRichiamata	La funzione di callback da chiamare quando l'utente riceve richieste di credenziali.
error	authErrorCallback	La funzione di callback da chiamare quando l'autenticazione fallisce.
success	authSuccessCallback	La funzione di callback da chiamare quando l'autenticazione ha successo.

AuthenticationErrorCode

Le AuthenticationError enumerazioni di codice disponibili nel modulo DCV

- INVALID_MESSAGE
- UNKNOWN_AUTH_MODE
- SESSION_NOT_AVAILABLE
- NO_SESSIONS
- WRONG_CREDENTIALS

- SASL_CHALLENGE
- SASL_AUTH_MECHANISM
- FAILED_COMMUNICATION
- AUTHENTICATION_REJECTED
- GENERIC_ERROR
- WRONG_CREDENTIALS_FORMAT
- WRONG_CREDENTIALS_TYPE
- UNREQUESTED_CREDENTIALS
- MISSING_CREDENTIAL

Type:

- numero

`authErrorCallback(authenticazione, errore)`

La funzione di callback da chiamare quando l'autenticazione fallisce.

Parametri:

Nome	Type (Tipo)	Descrizione						
<code>authentication</code>	Autenticazione	L'oggetto di autenticazione.						
<code>error</code>	Oggetto	L'oggetto di errore generato dal processo di autenticazione. <table border="1" data-bbox="1068 1522 1523 1866"> <thead> <tr> <th>Nome</th> <th>Type (Tipo)</th> <th>Descrizione</th> </tr> </thead> <tbody> <tr> <td><code>code</code></td> <td>AuthenticationErrorCode</td> <td>Il codice di errore.</td> </tr> </tbody> </table>	Nome	Type (Tipo)	Descrizione	<code>code</code>	AuthenticationErrorCode	Il codice di errore.
Nome	Type (Tipo)	Descrizione						
<code>code</code>	AuthenticationErrorCode	Il codice di errore.						

Nome	Type (Tipo)	Descrizione		
		Nome	Type (Tipo)	Descrizione
		message	string	Messaggio di errore.

authPromptCredentialsRichiamata (autenticazione, sfida)

La funzione di callback da chiamare quando l'utente riceve richieste di credenziali. L'utente deve rispondere alla sfida fornendo le credenziali richieste.

Parametri:

Nome	Type (Tipo)	Descrizione		
authentication	Autenticazione	L'oggetto di autenticazione.		
challenge	Oggetto	La sfida.		
		Nome	Type (Tipo)	Descrizione
		requires	Array. <Object>	Matrice di oggetti credenziali richiesti.

Nome	Type (Tipo)	Descrizione		
		Nome	Type (Tipo)	Descrizione
				<p data-bbox="1382 373 1511 478">Nome (Tipo)</p> <p data-bbox="1382 499 1593 814">Il nome della credenziale richiesta.</p> <p data-bbox="1382 877 1593 1192">Il tipo di credenziale richiesta.</p>

`authSuccessCallback(authenticazione, dati di autenticazione)`

La funzione di callback da chiamare quando l'autenticazione ha successo.

Parametri:

Nome	Type (Tipo)	Descrizione
<code>authentication</code>	Autenticazione	L'oggetto di autenticazione.
<code>authenticationData</code>	Array. <Object>	Una serie di oggetti che includono ID di sessione NICE

Nome	Type (Tipo)	Descrizione									
		DCV e token di autenticazione.									
		<table border="1"> <thead> <tr> <th>Nome</th> <th>Type (Tipo)</th> <th>Descrizione</th> </tr> </thead> <tbody> <tr> <td>sessionid</td> <td>string</td> <td>L'ID della sessione NICE DCV.</td> </tr> <tr> <td>authToken</td> <td>string</td> <td>Il token di autenticazione per la sessione NICE DCV.</td> </tr> </tbody> </table>	Nome	Type (Tipo)	Descrizione	sessionid	string	L'ID della sessione NICE DCV.	authToken	string	Il token di autenticazione per la sessione NICE DCV.
Nome	Type (Tipo)	Descrizione									
sessionid	string	L'ID della sessione NICE DCV.									
authToken	string	Il token di autenticazione per la sessione NICE DCV.									

Canale

I canali disponibili che possono essere specificati.

Type:

- «appunti» | «visualizzazione» | «input» | «audio» | «archiviazione file»

ChannelErrorCode

Le ChannelError enumerazioni di codice disponibili nel modulo DCV

- ALREADY_OPEN
- INITIALIZATION_FAILED

- REJECTED

Type:

- numero

`clipboardEventCallback(evento)`

La funzione di callback da chiamare quando `clipboardEvent` viene generato un.

Parametri:

Nome	Type (Tipo)	Descrizione			
event	Oggetto	Le informazioni sull'evento clipboard.			
		Nome	Type (Tipo)	Attribu	Descrizio
		name	stabilit		Sempre
					presenti.
			copia		Nome
					dell'even
			incolla		to.
			dataSi		
			lert		
			autoCr		
			one		
			newDa		
			ailable		
			autoPa		
			Done		
			Remoi		

Nome	Type (Tipo)	Descrizione			
		Nome	Type (Tipo)	Attribu	Descrizio ne
		or paste/ lableD			
		clippt Data	Ogget stringa		I dati negli appunti.
		autoC	boolea	<optio >	Indica se la copia automatic a dagli appunti della sessione agli appunti del client locale è abilitata .
		maxDa ze	numer	<optio >	La quantità massima di dati che

Nome	Type (Tipo)	Descrizione			
		Nome	Type (Tipo)	Attribu	Descrizio ne
					possono essere inseriti negli Appunti.
		error	string	<optio >	Informazi oni sull'erro re, se applicabi le.

ClosingReasonErrorCode

Le ClosingReasonError enumerazioni di codice disponibili nel modulo DCV

- TRANSPORT_ERROR
- NO_ERROR
- GENERIC_ERROR
- INTERNAL_SERVER_ERROR
- PROTOCOL_ERROR
- AUTHORIZATION_DENIED
- AUTHORIZATION_REVOKED
- ACCESS_REJECTED
- IDLE_TIMEOUT_EXPIRED
- DISCONNECT_BY_OWNER
- DISCONNECT_BY_USER
- EVICTED

Type:

- numero

Spazio cromatico

Gli spazi di colore disponibili che possono essere specificati.

Type:

- «RGB» | «YUV_REC601" | «YUV_REC709"

Richiamate di connessione

I callback disponibili per essere chiamati in caso di errore di connessione.

Type:

- Oggetto

Proprietà:

Nome	Type (Tipo)	Descrizione
disconnect	Disconnetti Callback	La funzione di callback da chiamare al termine della connessione.
displayLayout	displayLayoutCallback	La funzione di callback da richiamare quando si modifica il layout o la risoluzione del display.
displayAvailability	displayAvailabilityCallback	La funzione di callback da richiamare quando la disponibilità di un display cambia.
firstFrame	firstFrameCallback	La funzione di callback da chiamare quando il primo

Nome	Type (Tipo)	Descrizione
		frame viene ricevuto dal server NICE DCV.
filePrinted	filePrintedCallback	La funzione di callback da chiamare quando un file viene stampato sul server NICE DCV.
fileDownload	fileDownloadCallback	La funzione di callback da chiamare quando un file è pronto per essere scaricato dal server NICE DCV.
dataChannel	dataChannelCallback	La funzione di callback da chiamare quando il server NICE DCV invia una notifica sulla disponibilità di un canale dati.
licenseNotification	licenseNotificationCallback	La funzione di callback da chiamare quando il server NICE DCV invia una notifica sullo stato della licenza.
idleWarningNotification	idleWarningNotificationRichiamata	La funzione di callback da chiamare quando il server NICE DCV invia un avviso di timeout di inattività.
collaboratorList	collaboratorListCallback	La funzione di callback da chiamare quando il server NICE DCV invia l'elenco dei collaboratori (a partire dalla versione 1.1.0 di NICE DCV Web Client SDK).

Nome	Type (Tipo)	Descrizione
qualityIndicatorState	qualityIndicatorStateRichiamata	La funzione di callback da richiamare quando l'indicatore della qualità della connessione cambia stato.
filestorageEnabled	filestorageEnabledCallback	La funzione di callback da chiamare quando l'archiviazione dei file è abilitata o disabilitata.
featuresUpdate	featuresUpdateCallback	La funzione di callback da chiamare quando lo stato di una funzionalità cambia.
clipboardEvent	clipboardEventCallback	La funzione di callback da chiamare quando <code>clipboardEvent</code> viene generato un.
deviceChangeEvent	deviceChangeEventRichiamata	La funzione di callback da chiamare quando viene attivato un <code>deviceChangeEvent</code> .
screenshot	Screenshot Callback	La funzione di callback da chiamare quando <code>screenshot</code> è disponibile.

ConnectionConfig

La configurazione per una connessione NICE DCV.

Type:

- Oggetto

Proprietà:

Nome	Type (Tipo)	Descrizione
<code>url</code>	string	Il nome host e la porta del server NICE DCV in esecuzione nel seguente formato: <code>https://dcv_host_address:port</code> . Ad esempio: <code>https://my-dcv-server:8443</code> .
<code>sessionId</code>	string	L'ID della sessione NICE DCV.
<code>authToken</code>	string	Il token di autenticazione da utilizzare per la connessione al server.
<code>baseUrl</code>	string	L'URL assoluto o relativo da cui caricare i file SDK.
<code>resourceBaseUrl</code>	string	L'URL assoluto o relativo da cui accedere alle risorse DCV.
<code>enabledChannels</code>	Array. < Canale >	Indica l'elenco dei canali che possono essere abilitati. Se non viene specificato o viene fornito un array vuoto, per impostazione predefinita sono tutti i canali disponibili.
<code>losslessColorspace</code>	Spazio cromatico	Indica lo spazio cromatico che verrà utilizzato. Se non specificato, il valore predefinito viene, il valore predefinito viene, il valore predefinito viene, il valore predefinito viene.

Nome	Type (Tipo)	Descrizione
<code>divId</code>	string	L'ID dell' <code>div</code> oggetto nel DOM HTML in cui SDK deve creare la tela con il flusso remoto.
<code>volumeLevel</code>	integer	Il livello di volume preferito. L'intervallo valido è compreso tra 0 e 100.
<code>clipboardAutoSync</code>	booleano	Indica se la copia automatica dagli appunti della sessione NICE DCV agli appunti del client locale è abilitata per i browser Web compatibili.
<code>dynamicAudioTuning</code>	booleano	Indica se sintonizzare dinamicamente l'audio in base alle impostazioni audio del server NICE DCV quando viene stabilita una connessione.
<code>clientHiDpiScaling</code>	booleano	Indica se ridimensionare la tela in base al DPI del client.
<code>highColorAccuracy</code>	booleano	Indica se deve essere utilizzato a un'elevata precisione del colore, se disponibile. Se il valore non viene specificato, il valore impostato su <code>false</code> .
<code>enableWebCodecs</code>	Boolean	Indica se WebCodecs deve essere usato se disponibile. Se non specificato, viene usato il valore predefinito <code>false</code> .

Nome	Type (Tipo)	Descrizione
observers	Richiamate di connessione	Le funzioni di callback per richiamare eventi correlati alla connessione.
callbacks	Richiamate di connessione	Uguale allaobservers proprietà, ma ogni callback include l'oggetto Connection come primo parametro.

ConnectionErrorCode

Le ConnectionError enumerazioni di codice disponibili nel modulo DCV

- ALREADY_OPEN
- INVALID_CONFIG
- INITIALIZATION_FAILED
- REJECTED
- MAIN_CHANNEL_ALREADY_OPEN
- GENERIC_ERROR(a partire da DCV Server 2021.0)
- INTERNAL_SERVER_ERROR(a partire da DCV Server 2021.0)
- AUTHENTICATION_FAILED(a partire da DCV Server 2021.0)
- PROTOCOL_ERROR(a partire da DCV Server 2021.0)
- INVALID_SESSION_ID(a partire da DCV Server 2021.0)
- INVALID_CONNECTION_ID(a partire da DCV Server 2021.0)
- CONNECTION_LIMIT_REACHED(a partire da DCV Server 2021.0)
- SERVER_UNREACHABLE(a partire da DCV Server 2022.1)

Type:

- numero

CreateDirectory (percorso)

Parametri:

Nome	Type (Tipo)	Descrizione
path	string	Il percorso assoluto sul server in cui vogliamo creare una directory. Dovrebbe includere anche il nome della directory di destinazione.

CustomChannelErrorCode

Le CustomChannelError enumerazioni di codice disponibili nel modulo DCV

- TRANSPORT_ERROR

Type:

- numero

dataChannelCallback(informazioni)

La funzione di callback da chiamare quando il server NICE DCV invia una notifica sulla disponibilità di un canale dati.

Parametri:

Nome	Type (Tipo)	Descrizione
info	Oggetto	Informazioni sul canale dati.

Nome	Type (Tipo)	Descrizione									
		<table border="1"> <thead> <tr> <th>Nome</th> <th>Type (Tipo)</th> <th>Descrizione</th> </tr> </thead> <tbody> <tr> <td>name</td> <td>string</td> <td>Il nome del canale dati.</td> </tr> <tr> <td>token</td> <td>string</td> <td>Il token di autenticazione per il canale di dati.</td> </tr> </tbody> </table>	Nome	Type (Tipo)	Descrizione	name	string	Il nome del canale dati.	token	string	Il token di autenticazione per il canale di dati.
Nome	Type (Tipo)	Descrizione									
name	string	Il nome del canale dati.									
token	string	Il token di autenticazione per il canale di dati.									

deleteFile (percorso)

Parametri:

Nome	Type (Tipo)	Descrizione
path	string	Il percorso assoluto sul server che identifica il file che vogliamo eliminare.

deviceChangeEventRichiamata ()

La funzione di callback da chiamare quando viene attivato un deviceChange evento.

DisconnectCallback (motivo)

La funzione di callback da chiamare al termine della connessione.

Parametri:

Nome	Type (Tipo)	Descrizione									
reason	Oggetto	Il motivo della disconnessione.									
		<table border="1"> <thead> <tr> <th>Nome</th> <th>Type (Tipo)</th> <th>Descrizione</th> </tr> </thead> <tbody> <tr> <td>code</td> <td>numero</td> <td>Il codice del motivo.</td> </tr> <tr> <td>message</td> <td>string</td> <td>Il messaggio del motivo.</td> </tr> </tbody> </table>	Nome	Type (Tipo)	Descrizione	code	numero	Il codice del motivo.	message	string	Il messaggio del motivo.
Nome	Type (Tipo)	Descrizione									
code	numero	Il codice del motivo.									
message	string	Il messaggio del motivo.									

`displayAvailabilityCallback(stato, DisplayID)`

La funzione di callback da richiamare quando la disponibilità di un display cambia.

Parametri:

Nome	Type (Tipo)	Descrizione						
status	Oggetto	Lo stato del display.						
		<table border="1"> <thead> <tr> <th>Nome</th> <th>Type (Tipo)</th> <th>Descrizione</th> </tr> </thead> <tbody> <tr> <td>enablec</td> <td>booleanc</td> <td>Indica se il display è</td> </tr> </tbody> </table>	Nome	Type (Tipo)	Descrizione	enablec	booleanc	Indica se il display è
Nome	Type (Tipo)	Descrizione						
enablec	booleanc	Indica se il display è						

Nome	Type (Tipo)	Descrizione		
		Nome	Type (Tipo)	Descrizione
				abilitato
				.
		closed	booleanc	Indica se il display è chiuso.
displayId	numero	L'identificatore del display.		

DisplayConfigErrorCode

Le DisplayConfigError enumerazioni di codice disponibili nel modulo DCV

- INVALID_ARGUMENT
- UNSUPPORTED_OPERATION
- NO_CHANNEL

Type:

- numero

`displayLayoutCallback(larghezza del server, altezza del server, testate)`

La funzione di callback da richiamare quando si modifica il layout o la risoluzione del display.

Parametri:

Nome	Type (Tipo)	Descrizione
serverWidth	numero	La larghezza (in pixel) dello schermo principale.
serverHeight	numero	L'altezza (in pixel) del display principale.
heads	Array. < Monitora >	Le testine di visualizzazione sono supportate dal server NICE DCV.

caratteristica

I valori delle funzionalità.

- `display`- Indica la disponibilità di un flusso video a schermo singolo.
- `display-multi`- Indica la disponibilità di un flusso video con più display.
- `high-color-accuracy`- Indica la disponibilità di un'elevata precisione del colore (a partire dalla versione NICE DCV Web Client SDK 1.1.0).
- `mouse`- Indica la disponibilità della funzionalità del mouse.
- `keyboard`- Indica la disponibilità della funzionalità della tastiera.
- `keyboard-sas`- Indica la disponibilità della funzionalità della sequenza SAS (Control + Alt + Delete).
- `relative-mouse`- Indica la disponibilità della modalità relativa del mouse.
- `clipboard-copy`- Indica la disponibilità della funzionalità di copia degli appunti dal server NICE DCV al client.
- `clipboard-paste`- Indica la disponibilità della funzionalità di incollaggio degli appunti dal client al server NICE DCV.
- `audio-in`- Indica la disponibilità della funzionalità di ingresso audio tramite il microfono.
- `audio-out`- Indica la disponibilità della funzionalità di riproduzione audio.
- `webcam`- Indica la disponibilità della funzionalità di streaming della webcam.
- `file-download`- Indica la disponibilità della funzionalità di download dei file dal server NICE DCV al client.

- `file-upload`- Indica la disponibilità della funzionalità di caricamento dei file dal client al server NICE DCV.
- `timezone-redirect`- Indica la disponibilità della funzionalità di reindirizzamento del fuso orario (a partire dalla versione 1.3.0 di NICE DCV Web Client SDK).

Type:

- `string`

`featuresUpdateCallback`(Elenco delle caratteristiche)

La funzione di callback da chiamare quando lo stato di una funzionalità cambia.

Parametri:

Nome	Type (Tipo)	Descrizione
<code>featuresList</code>	Array. < funzionalità >	Una serie di funzionalità che sono state modificate.

`fileDownloadCallback`(Risorsa di file)

La funzione di callback da chiamare quando un file è pronto per essere scaricato dal server NICE DCV.

Parametri:

Nome	Type (Tipo)	Descrizione
<code>fileResource</code>	Oggetto	Informazioni sul file pronto per il download.

Nome	Type (Tipo)	Descrizione		
		Nome	Type (Tipo)	Descrizione
		id	string	L'identificatore del file.
		url	string	L'URL da utilizzare per scaricare il file.
		domain	string	Il dominio delle risorse.
		token	string	Il token di autenticazione da utilizzare per scaricare il file. Il token è incluso anche nell'URL.

filePrintedCallback(Risorsa di stampa)

La funzione di callback da chiamare quando un file viene stampato sul server NICE DCV.

Parametri:

Nome	Type (Tipo)	Descrizione															
<code>printResource</code>	Oggetto	<p>Informazioni sul file stampato.</p> <table border="1"> <thead> <tr> <th>Nome</th> <th>Type (Tipo)</th> <th>Descrizione</th> </tr> </thead> <tbody> <tr> <td><code>id</code></td> <td><code>string</code></td> <td>L'identificatore del file stampato.</td> </tr> <tr> <td><code>url</code></td> <td><code>string</code></td> <td>L'URL da utilizzare per scaricare il file stampato.</td> </tr> <tr> <td><code>domain</code></td> <td><code>string</code></td> <td>Il dominio delle risorse. In questo caso, <code>printer</code>.</td> </tr> <tr> <td><code>token</code></td> <td><code>string</code></td> <td>Il token di autenticazione</td> </tr> </tbody> </table>	Nome	Type (Tipo)	Descrizione	<code>id</code>	<code>string</code>	L'identificatore del file stampato.	<code>url</code>	<code>string</code>	L'URL da utilizzare per scaricare il file stampato.	<code>domain</code>	<code>string</code>	Il dominio delle risorse. In questo caso, <code>printer</code> .	<code>token</code>	<code>string</code>	Il token di autenticazione
Nome	Type (Tipo)	Descrizione															
<code>id</code>	<code>string</code>	L'identificatore del file stampato.															
<code>url</code>	<code>string</code>	L'URL da utilizzare per scaricare il file stampato.															
<code>domain</code>	<code>string</code>	Il dominio delle risorse. In questo caso, <code>printer</code> .															
<code>token</code>	<code>string</code>	Il token di autenticazione															

Nome	Type (Tipo)	Descrizione		
		Nome	Type (Tipo)	Descrizione
				da utilizzare per scaricare il file stampato. Il token è incluso anche nell'URL.

archiviazione di file

Oggetto che consente di esplorare ed eseguire azioni sul file system.

Type:

- Oggetto

Proprietà:

Nome	Type (Tipo)	Descrizione
list	elenco	Funzione che permette di elencare gli elementi (file e directory) presenti nel percorso fornito sul server.
createDirectory	Crea cartella	Funzione che consente di creare una directory nel

Nome	Type (Tipo)	Descrizione
		percorso specificato sul server.
retrieveFile	Recupera file	Funzione che consente di scaricare localmente un file nel percorso specificato sul server.
deleteFile	Elimina file	Funzione che consente di eliminare un file nel percorso specificato sul server.
renameFile	Rinomina file	Funzione che consente di rinominare un file dal percorso di origine specificato al percorso di destinazione specificato.
renameDirectory	Rinomina cartella	Funzione che consente di rinominare una directory dal percorso di origine specificato al percorso di destinazione assoluto.
storeFile	Memorizza file	Funzione che permette di caricare un file locale nel percorso fornito sul server.

filestorageEnabledCallback(abilitato)

La funzione di callback da chiamare quando l'archiviazione dei file è abilitata. Canale Lazy solo su Internet Explorer 11.

Parametri:

Nome	Type (Tipo)	Descrizione
<code>enabled</code>	booleano	Indica se la memorizzazione dei file è abilitata.

FileStorageErrorCode

Le FileStorageError enumerazioni di codice disponibili nel modulo DCV

- CANCELLED
- ABORTED
- INVALID_ARGUMENT
- NOT_IMPLEMENTED
- ERROR
- ALREADY_EXIST
- NOT_FOUND

Type:

- numero

firstFrameCallback(Ridimensiona relativeMouseMode abilitato, DisplayID)

La funzione di callback da chiamare quando il primo frame viene ricevuto dal server NICE DCV. Emesso per ogni display.

Parametri:

Nome	Type (Tipo)	Descrizione
<code>resizeEnabled</code>	booleano	Indica se il server supporta il ridimensionamento del layout di visualizzazione del client.

Nome	Type (Tipo)	Descrizione
<code>relativeMouseModeEnabled</code>	booleano	Indica se il server supporta la modalità mouse relativa.
<code>displayId</code>	numero	L'identificatore del display.

`idleWarningNotification`Richiamata (`disconnectionDateTime`)

La funzione di callback da chiamare quando il server NICE DCV invia un avviso di timeout di inattività.

Parametri:

Nome	Type (Tipo)	Descrizione
<code>disconnectionDateTime</code>	Data	La data e l'ora della disconnessione.

`collaboratorList`Callback(`collaboratori`)

La funzione di callback da chiamare quando il server NICE DCV invia l'elenco dei collaboratori.

Parametri:

Nome	Type (Tipo)	Descrizione						
<code>collaborators</code>	Array. <Object>	Un elenco di oggetti contenenti informazioni sui collaboratori. <table border="1" data-bbox="1068 1602 1511 1885"> <thead> <tr> <th>Nome</th> <th>Type (Tipo)</th> <th>Descrizione</th> </tr> </thead> <tbody> <tr> <td><code>username</code></td> <td>string</td> <td>Il nome utente del</td> </tr> </tbody> </table>	Nome	Type (Tipo)	Descrizione	<code>username</code>	string	Il nome utente del
Nome	Type (Tipo)	Descrizione						
<code>username</code>	string	Il nome utente del						

Nome	Type (Tipo)	Descrizione		
		Nome	Type (Tipo)	Descrizione
				collaboratore.
		owner	booleano	Indica se il collaboratore è il proprietario della sessione.
		connect nId	numero	Indica l'ID assegnato dal server alla connessione.

licenseNotificationCallback(notifica)

La funzione di callback da chiamare quando il server NICE DCV invia una notifica sullo stato della licenza.

Parametri:

Nome	Type (Tipo)	Descrizione
notification	Oggetto	La notifica.

Nome	Type (Tipo)	Descrizione																					
		<table border="1"> <thead> <tr> <th>Nome</th> <th>Type (Tipo)</th> <th>Descrizione</th> </tr> </thead> <tbody> <tr> <td>product</td> <td>string</td> <td>Il prodotto DCV.</td> </tr> <tr> <td>status</td> <td>string</td> <td>Lo stato della licenza.</td> </tr> <tr> <td>message</td> <td>string</td> <td>Un messaggio.</td> </tr> <tr> <td>leftDay</td> <td>numero</td> <td>Il numero di giorni prima della scadenza della licenza.</td> </tr> <tr> <td>isDemo</td> <td>booleanc</td> <td>Indica se la licenza è una licenza demo.</td> </tr> <tr> <td>numUnlicensed</td> <td>numero</td> <td>Il numero di</td> </tr> </tbody> </table>	Nome	Type (Tipo)	Descrizione	product	string	Il prodotto DCV.	status	string	Lo stato della licenza.	message	string	Un messaggio.	leftDay	numero	Il numero di giorni prima della scadenza della licenza.	isDemo	booleanc	Indica se la licenza è una licenza demo.	numUnlicensed	numero	Il numero di
Nome	Type (Tipo)	Descrizione																					
product	string	Il prodotto DCV.																					
status	string	Lo stato della licenza.																					
message	string	Un messaggio.																					
leftDay	numero	Il numero di giorni prima della scadenza della licenza.																					
isDemo	booleanc	Indica se la licenza è una licenza demo.																					
numUnlicensed	numero	Il numero di																					

Nome	Type (Tipo)	Descrizione		
		Nome	Type (Tipo)	Descrizione
				connessioni senza licenza.
		licensi Mode	string	La modalità di licenza.
		documentationUrl	string	L'URL della documentazione.

elenco (percorso)

Parametri:

Nome	Type (Tipo)	Descrizione
path	string	Il percorso assoluto sul server di cui vogliamo elencare il contenuto.

LogLevel

I livelli di registro SDK disponibili.

Type:

- TRACCIA | DEBUG | INFORMAZIONI | AVVISO | ERRORE | SILENZIOSO

Monitor (Monitoraggio)

Type:

- Oggetto

Proprietà:

Nome	Type (Tipo)	Descrizione									
name	string	Il nome della testina del display.									
rect	Oggetto	Informazioni sulla testina del display. <table border="1"> <thead> <tr> <th>Nome</th> <th>Type (Tipo)</th> <th>Descrizione</th> </tr> </thead> <tbody> <tr> <td>x</td> <td>numero</td> <td>Lax coordinata iniziale della testina del display.</td> </tr> <tr> <td>y</td> <td>numero</td> <td>Lay coordinata iniziale della testina del display.</td> </tr> </tbody> </table>	Nome	Type (Tipo)	Descrizione	x	numero	Lax coordinata iniziale della testina del display.	y	numero	Lay coordinata iniziale della testina del display.
Nome	Type (Tipo)	Descrizione									
x	numero	Lax coordinata iniziale della testina del display.									
y	numero	Lay coordinata iniziale della testina del display.									

Nome	Type (Tipo)	Descrizione		
		Nome	Type (Tipo)	Descrizione
		width	numero	La larghezza (in pixel) della testina del display.
		height	numero	L'altezza (in pixel) della testina del display.
primary	booleano	Indica se la testina dello schermo è la testina di visualizzazione principale. Questo viene determinato dal sistema operativo remoto, se disponibile.		
dpi	numero	Il DPI della testina del display.		

MultiMonitorErrorCode

Le MultiMonitorError enumerazioni di codice disponibili nel modulo DCV

- NO_DISPLAY_CHANNEL
- MAX_DISPLAY_NUMBER_REACHED

- INVALID_ARGUMENT
- DISPLAY_NOT_OPENED_BY_SERVER
- REQUEST_TIMEOUT
- GENERIC_ERROR
- NO_ERROR

Type:

- numero

qualityIndicatorStateCallback (stato)

La funzione di callback da richiamare quando l'indicatore della qualità della connessione cambia stato.

Parametri:

Nome	Type (Tipo)	Descrizione												
state	Array. <Object>	Informazioni sulla qualità della connessione.												
		<table border="1"> <thead> <tr> <th>Nome</th> <th>Type (Tipo)</th> <th>Descrizione</th> </tr> </thead> <tbody> <tr> <td>name</td> <td>string</td> <td>Il nome dell'indicatore.</td> </tr> <tr> <td>status</td> <td>NORMAL AVVISO CRITICO</td> <td>Descrizione dello stato.</td> </tr> <tr> <td>change</td> <td>booleano</td> <td>Indica se lo</td> </tr> </tbody> </table>	Nome	Type (Tipo)	Descrizione	name	string	Il nome dell'indicatore.	status	NORMAL AVVISO CRITICO	Descrizione dello stato.	change	booleano	Indica se lo
Nome	Type (Tipo)	Descrizione												
name	string	Il nome dell'indicatore.												
status	NORMAL AVVISO CRITICO	Descrizione dello stato.												
change	booleano	Indica se lo												

Nome	Type (Tipo)	Descrizione		
		Nome	Type (Tipo)	Descrizione
				stato è cambiato.

Rinomina cartella (src, dest)

Parametri:

Nome	Type (Tipo)	Descrizione
src	string	Il percorso di origine assoluto sul server che identifica la directory che vogliamo rinominare.
dest	string	Il percorso di destinazione assoluto sul server che specifica il percorso di destinazione e il nome della directory.

Rinomina file (src, dest)

Parametri:

Nome	Type (Tipo)	Descrizione
src	string	Il percorso di origine assoluto sul server che identifica il file che vogliamo rinominare.
dest	string	Il percorso di destinazione assoluto sul server

Nome	Type (Tipo)	Descrizione
		che specifica il percorso di destinazione e il nome del file.

ResolutionErrorCode

Le ResolutionError enumerazioni di codice disponibili nel modulo DCV

- INVALID_ARGUMENT
- NO_CHANNEL
- NOT_IMPLEMENTED

Type:

- numero

RetrieveFile (percorso)

Parametri:

Nome	Type (Tipo)	Descrizione
path	string	Il percorso assoluto sul server che identifica il file che vogliamo scaricare localment e.

ScreenshotCallback (immagine dello schermo)

La funzione di callback da chiamare quando è disponibile uno screenshot.

Parametri:

Nome	Type (Tipo)	Descrizione
screenshot	byte[]	Buffer di screenshot in formato PNG <code>onull</code> se il recupero dello screenshot non è riuscito.

ScreenshotErrorCode

Le ScreenshotError enumerazioni di codice disponibili nel modulo DCV

- NO_CHANNEL
- GENERIC_ERROR

Type:

- numero

Informazioni sul server

Type:

- Oggetto

Proprietà:

Nome	Type (Tipo)	Descrizione
name	string	Il nome del software.
version	Oggetto	Il numero di versione del software.

Nome	Type (Tipo)	Descrizione												
		<table border="1"> <thead> <tr> <th data-bbox="1068 226 1214 352">Nome</th> <th data-bbox="1214 226 1360 352">Type (Tipo)</th> <th data-bbox="1360 226 1513 352">Descrizione</th> </tr> </thead> <tbody> <tr> <td data-bbox="1068 352 1214 674">major</td> <td data-bbox="1214 352 1360 674">numero</td> <td data-bbox="1360 352 1513 674">Il numero di versione principale.</td> </tr> <tr> <td data-bbox="1068 674 1214 947">minor</td> <td data-bbox="1214 674 1360 947">numero</td> <td data-bbox="1360 674 1513 947">Il numero di versione minore.</td> </tr> <tr> <td data-bbox="1068 947 1214 1304">revision</td> <td data-bbox="1214 947 1360 1304">numero</td> <td data-bbox="1360 947 1513 1304">Il numero di versione della revisione.</td> </tr> </tbody> </table>	Nome	Type (Tipo)	Descrizione	major	numero	Il numero di versione principale.	minor	numero	Il numero di versione minore.	revision	numero	Il numero di versione della revisione.
Nome	Type (Tipo)	Descrizione												
major	numero	Il numero di versione principale.												
minor	numero	Il numero di versione minore.												
revision	numero	Il numero di versione della revisione.												
os	string	Il sistema operativo.												
arch	string	L'architettura.												
hostname	string	Il nome host.												

stats

Type:

- Oggetto

Proprietà:

Nome	Type (Tipo)	Descrizione
fps	numero	I fotogrammi correnti al secondo.
traffic	numero	Il traffico corrente in bit/s.
peakTraffic	numero	Il picco di traffico in bit/s da quando è stata stabilita la connessione.
latency	numero	Latenza corrente in ms.
currentChannels	numero	Il numero di canali aperti da quando è stata stabilita la connessione.
openedChannels	numero	Il numero di canali attualmente aperti.
channelErrors	numero	Il numero di canali che hanno segnalato un errore.

StoreFile (file, directory)

Parametri:

Nome	Type (Tipo)	Descrizione
file	File	L'oggetto del file (per maggiori informazioni vedi https://developer.mozilla.org/en-US/docs/Web/API/File) che vogliamo caricare sul server.

Nome	Type (Tipo)	Descrizione
<code>dir</code>	string	Il percorso assoluto sul server in cui vogliamo caricare il file.

TimezoneRedirectionErrorCode

Le `TimezoneRedirectionError` enumerazioni di codice disponibili nel modulo DCV

- `INVALID_ARGUMENT`
- `NO_CHANNEL`
- `USER_CANNOT_CHANGE`

Type:

- numero

TimezoneRedirectionSettingCode

Le `TimezoneRedirectionSetting` enumerazioni di codice disponibili nel modulo DCV

- `ALWAYS_OFF`
- `ALWAYS_ON`
- `CLIENT_DECIDES`

Type:

- numero

TimezoneRedirectionStatusCode

Le `TimezoneRedirectionStatus` enumerazioni di codice disponibili nel modulo DCV

- `SUCCESS`
- `PERMISSION_ERROR`
- `GENERIC_ERROR`

Type:

- numero

WebcamErrorCode

Le WebcamError enumerazioni di codice disponibili nel modulo DCV

- SETTING_WEBCAM_FAILED
- CHANNEL_NOT_AVAILABLE

Type:

- numero

Classe di connessione

La classe di connessione ottenuta chiamando il [connect metodo](#) del dcv modulo. Per un esempio che mostra come usarlo, consulta la sezione [Guida introduttiva](#).

Esponde

- [Metodi](#)

Metodi

Elenco

- [attachDisplay \(win, displayConf\) → {Promessa. <number>|Promessa. < {code: MultiMonitorErrorCode, messaggio: string} >}](#)
- [captureClipboardEvents\(abilitato, win, displayID\) → {void}](#)
- [DetachDisplay \(displayID\) → {void}](#)
- [disconnetti \(\) → {void}](#)
- [Disconnetti Collaborator \(connectionId\) → {void}](#)
- [enableDisplayQualityAggiornamenti \(abilita\) → {void}](#)
- [enableTimezoneRedirection\(abilita\) → {Promessa|Promessa. < {code: TimezoneRedirectionErrorCode, messaggio: string} >}](#)

- [enterRelativeMouseModalità \(\) → {void}](#)
- [getConnectedDevices\(\) → {Promessa. <Matrice. <MediaDeviceInfo >>|Promessa. < {message: string} >}](#)
- [getFileExplorer\(\) → {Promessa. < archiviazione dei file >|Promessa. < {code: ChannelErrorCode, messaggio: string} >}](#)
- [getServerInfo\(\) → {Informazioni sul server}](#)
- [getScreenshot \(\) → {Promessa|Promessa. < {code: ScreenshotErrorCode, messaggio: string} >}](#)
- [getStats \(\) → {statistiche}](#)
- [latchModifierKey\(chiave, posizione, isDown\) → {boolean}](#)
- [OpenChannel \(nome, authToken, callback, namespace\) → {Promise|Promessa. < {code: ChannelErrorCode, messaggio: string} >}](#)
- [QueryFeature \(FeatureName\) → {Promessa. < {abilitato: booleano, remoto? : stringa, AutoCopy? : booleano, autoPaste? : booleano, ServiceStatus? : stringa, disponibile? : boolean} >|Promessa. < {message: string} >}](#)
- [registerKeyboardShortcuts\(scelte rapide\) → {void}](#)
- [requestDisplayConfig\(highColorAccuracy\) → {Promessa|Promessa. < {code: DisplayConfigErrorCode, messaggio: string} >}](#)
- [requestDisplayLayout\(layout\) → {Promessa|Promessa. < {code: ResolutionErrorCode, messaggio: string} >}](#)
- [RequestResolution \(larghezza, altezza\) → {Promessa | Promessa. < {code: ResolutionErrorCode, messaggio: string} >}](#)
- [sendKeyboardEvent\(evento\) → {booleano}](#)
- [sendKeyboardShortcut\(scorciatoia\) → {void}](#)
- [setDisplayQuality\(min, max opt\) → {void}](#)
- [setDisplayScale\(ScaleRatio, DisplayID\) → {Promessa | Promessa. < {code: ResolutionErrorCode, message: string} >} \(OBSOLETO\)](#)
- [setKeyboardQuirks\(stranezze\) → {void}](#)
- [setMaxDisplayRisoluzione \(larghezza massima, altezza massima\) → {void}](#)
- [setMicrophone \(abilita\) → {Promessa|Promessa. < {code: AudioErrorCode, messaggio: string} >}](#)
- [setMinDisplayRisoluzione \(larghezza minima, altezza minima\) → {void}](#)
- [setUploadBandwidth\(valore\) → {numero}](#)
- [setVolume \(volume\) → {void}](#)

- [setWebcam \(abilita, ID dispositivo\) → {Promessa | Promessa. < {code: WebcamErrorCode, messaggio: string} >}](#)
- [syncClipBoards \(\) → {booleano}](#)

[attachDisplay \(win, displayConf\) → {Promessa. <number>|Promessa. < {code: MultiMonitorErrorCode, messaggio: string} >}](#)

Collega uno schermo specifico a una finestra. Non è possibile collegare il display principale. Se eseguito correttamente, la funzione restituisce `displayId`.

Parametri:

Nome	Type (Tipo)	Descrizione			
<code>win</code>	Oggetto	La finestra a cui deve essere collegato il display.			
<code>displayConf</code>	Oggetto	La configurazione del display.			
		Nome	Type (Tipo)	Attribu	Descrizio
		<code>displ</code>	numer	<optio	L'ID del display.
		<code>displ</code>	<code>vName</code>		Il nome del div dello schermo.

Valori restituiti:

Promessa. Se rifiutata, la promessa restituisce un oggetto di errore.

Type (Tipo)

Promessa. <number>| Promessa. < {code: [MultiMonitorErrorCode](#), messaggio: stringa} >

`captureClipboardEvents(abilitato, win, displayID) → {void}`

Avvia o interrompe l'ascolto degli eventi di copia-incolla. Nel caso degli appunti interattivi (sempre nel caso di incolla) dobbiamo iniziare ad ascoltare gli eventi di copia/incolla. Potrebbe essere utile avviare e interrompere l'ascolto solo quando è necessario, ad esempio quando viene mostrata una modalità.

Parametri:

Nome	Type (Tipo)	Attributi	Descrizione
<code>enabled</code>	booleano		Per iniziare ad ascoltare gli eventi, specificat <code>true</code> . Per interrompere l'ascolto degli eventi, specifica <code>false</code> .
<code>win</code>	Oggetto	<optional>	La finestra in cui ascoltare gli eventi. Se omesso, viene utilizzata la finestra predefinita.
<code>displayId</code>	numero	<optional>	L'ID del display che dovrebbe ascoltare gli eventi. Se omesso, viene utilizzata la visualizzazione predefinita della finestra.

Valori restituiti:

Type (Tipo)

void

DetachDisplay (displayID) → {void}

Scollega un display specifico. Il display principale non può essere rimosso.

Parametri:

Nome	Type (Tipo)	Descrizione
displayId	numero	L'ID del display da scollegare.

Valori restituiti:

Type (Tipo)

void

disconnetti () → {void}

Si disconnette dal server NICE DCV e chiude la connessione.

Valori restituiti:

Type (Tipo)

void

Disconnetti Collaborator (connectionId) → {void}

Richiede la disconnessione del collaboratore connesso con l'ID di connessione fornito (a partire dalla versione 1.1.0 di NICE DCV Web Client SDK).

Parametri:

Nome	Type (Tipo)	Descrizione
connectionId	booleano	L'id della connessione che verrà disconnessa.

Valori restituiti:

Type (Tipo)

void

enableDisplayQualityAggiornamenti (abilita) → {void}

Abilita o disabilita gli aggiornamenti della qualità dello schermo per le aree di streaming che non ricevono aggiornamenti. La disattivazione degli aggiornamenti della qualità del display riduce l'utilizzo della larghezza di banda, ma riduce anche la qualità dello schermo.

Parametri:

Nome	Type (Tipo)	Descrizione
enable	booleano	Per abilitare gli aggiornamenti della qualità del display, specificare <code>true</code> . Per disabilitare gli aggiornamenti della qualità del display, specifica <code>false</code> .

Valori restituiti:

Type (Tipo)

void

`enableTimezoneRedirection(abilita) → {Promessa|Promessa. < {code: TimezoneRedirectionErrorCode, messaggio: string} >}`

Consente di attivare o disattivare il reindirizzamento del fuso orario. Una volta abilitato, il client richiede al server di fare in modo che il fuso orario del desktop del server corrisponda al fuso orario del client.

Parametri:

Nome	Type (Tipo)	Descrizione
<code>enable</code>	booleano	Per abilitare il reindirizzamento del fuso orario, specifica <code>true</code> . Per disabilitare il reindirizzamento del fuso orario, specificare <code>false</code> .

Valori restituiti:

Promessa. Se rifiutata, la promessa restituisce un oggetto di errore.

Type (Tipo)

Promessa. <number>| Promessa. < {code: [TimezoneRedirectionErrorCode](#), messaggio: stringa} >

`enterRelativeMouseModalità () → {void}`

Abilita la modalità relativa del mouse.

Valori restituiti:

Type (Tipo)

void

`getConnectedDevices() → {Promessa. <Matrice. <MediaDeviceInfo >>|Promessa. < {message: string} >}`

Richiede un elenco dei dispositivi multimediali collegati al computer client.

Valori restituiti:

In caso di successo, restituisce una promessa che si risolve in una serie di `MediaDeviceInfo` oggetti. Per ulteriori informazioni,, consulta <https://developer.mozilla.org/en-US/docs/Web/API/MediaDeviceInfo>. Se rifiutata, la promessa restituisce un oggetto di errore.

Type (Tipo)

Promessa. <Matrice. <`MediaDeviceInfo` >> | Promessa. < {message: string} >

`getFileExplorer()` → {Promessa. < [archiviazione dei file](#) >|Promessa. < {code: [ChannelErrorCode](#), messaggio: string} >}

Ottiene un oggetto per gestire l'archiviazione dei file del server NICE DCV.

Valori restituiti:

Promessa. Si risolve nell'oggetto File Explorer se soddisfatto o in un oggetto di errore se rifiutato.

Type (Tipo)

Promessa. < [filestorage](#) > | Promessa. < {code: [ChannelErrorCode](#), messaggio: stringa} >

`getServerInfo()` → {[Informazioni sul server](#)}

Restituisce informazioni sul server NICE DCV.

Valori restituiti:

Informazioni sul software del server.

Type (Tipo)

[Informazioni sul server](#)

`getScreenshot ()` → {Promessa|Promessa. < {code: [ScreenshotErrorCode](#), messaggio: string} >}

Recupera lo screenshot del desktop remoto in formato PNG. Lo screenshot verrà restituito nell'[osservatore ScreenshotCallback](#). null verranno invece restituiti in caso di guasti.

Valori restituiti:

Promessa che si risolve se la richiesta viene elaborata. Se rifiutato, riceviamo un oggetto di errore.

Type (Tipo)

Promessa | Promessa. < {code: [ScreenshotErrorCode](#), messaggio: stringa} >

getStats () → [{statistiche}](#)

Ottiene statistiche sul server NICE DCV.

Valori restituiti:

Informazioni sulle statistiche di streaming.

Type (Tipo)

[statistiche](#)

latchModifierKey(chiave, posizione, isDown) → {boolean}

Invia una singola tastierakeydown okeyup evento per un modificatore consentito.

Parametri:

Nome	Type (Tipo)	Descrizione
key	Controllo Alt AltGraph Meta Sistema operativo Shift	La chiave da inviare.
location	KeyboardEvent.posizione	La posizione della chiave. Per ulteriori informazioni,, consulta https://developer.mozilla.org/en-US/docs/Web/API/KeyboardEvent/location .

Nome	Type (Tipo)	Descrizione
isDown	booleano	Se l'evento chiave da inserire è un keydown (<code>true</code>) o un keyup (<code>false</code>).

Valori restituiti:

Se la combinazione richiesta è valida, la funzione restituisce `true`, altrimenti `false`.

Type (Tipo)

booleano

OpenChannel (nome, authToken, callback, namespace) → {Promise|Promessa. < {code: [ChannelErrorCode](#), messaggio: string} >}

Apri un canale dati personalizzato sulla connessione se è stato creato sul server NICE DCV.

Parametri:

Nome	Type (Tipo)	Descrizione
name	string	Nome del canale.
authToken	string	Il token di autenticazione da utilizzare per connettersi al canale.
callbacks	Oggetto	Le funzioni di callback <code>onMessage</code> e <code>onClose</code> da chiamare.
namespace	string	Namespace del canale. Disponibile a partire da NICE DCV Web Client SDK 1.2.0 e NICE DCV Server 2022.1.

Valori restituiti:

Promessa. Se rifiutato, riceviamo un oggetto di errore.

Type (Tipo)

Promessa | Promessa. < {code: [ChannelErrorCode](#), messaggio: stringa} >

QueryFeature (FeatureName) → {Promessa. < {abilitato: booleano, remoto? : stringa, AutoCopy? : booleano, autoPaste? : booleano, ServiceStatus? : stringa, disponibile? : boolean} > | Promessa. < {message: string} >}

Richiede lo stato di una funzionalità specifica del server NICE DCV.

Parametri:

Nome	Type (Tipo)	Descrizione
featureName	caratteristica	Il nome della funzionalità da eseguire la query.

Valori restituiti:

Promessa. Se risolta, la funzione restituisce un `status` oggetto che contiene sempre una `enabled` proprietà e possibilmente anche altre proprietà. Se rifiutata, la funzione restituisce un `error` oggetto.

Type (Tipo)

{Promessa. < {abilitato: booleano, remoto? : stringa, AutoCopy? : booleano, autoPaste? : booleano, ServiceStatus? : stringa, disponibile? : boolean} > | Promessa. < {message: string} >

registerKeyboardShortcuts(scelte rapide) → {void}

Consente di registrare le scelte rapide da tastiera.

Parametri:

Nome	Type (Tipo)	Descrizione												
shortcuts	Array. <Object>	<p>La serie di chiavi e mappature da registrare.</p> <table border="1"> <thead> <tr> <th>Nome</th> <th>Type (Tipo)</th> <th>Descrizione</th> </tr> </thead> <tbody> <tr> <td>sequence</td> <td>Array. <Object></td> <td> <p>La scorciatoia da tastiera per la registrazione.</p> <table border="1"> <thead> <tr> <th>Nome</th> <th>Type (Tipo)</th> <th>Descrizione</th> </tr> </thead> <tbody> <tr> <td>keyCode</td> <td>KeyboardEvent.keyCode</td> <td> <p>Il valore del tasto premuto dall'utente. Per ulteriori informazioni,,, consulta <a 102="" 55="" 920="" 936"="" data-label="Page-Footer" href="https://developer.</p> </td> </tr> </tbody> </table> </td> </tr> </tbody> </table> </td> </tr> </tbody> </table> </div> <div data-bbox="> <p>Metodi</p> </p></td></tr></tbody></table></td></tr></tbody></table>	Nome	Type (Tipo)	Descrizione	sequence	Array. <Object>	<p>La scorciatoia da tastiera per la registrazione.</p> <table border="1"> <thead> <tr> <th>Nome</th> <th>Type (Tipo)</th> <th>Descrizione</th> </tr> </thead> <tbody> <tr> <td>keyCode</td> <td>KeyboardEvent.keyCode</td> <td> <p>Il valore del tasto premuto dall'utente. Per ulteriori informazioni,,, consulta <a 102="" 55="" 920="" 936"="" data-label="Page-Footer" href="https://developer.</p> </td> </tr> </tbody> </table> </td> </tr> </tbody> </table> </td> </tr> </tbody> </table> </div> <div data-bbox="> <p>Metodi</p> </p></td></tr></tbody></table>	Nome	Type (Tipo)	Descrizione	keyCode	KeyboardEvent.keyCode	<p>Il valore del tasto premuto dall'utente. Per ulteriori informazioni,,, consulta <a 102="" 55="" 920="" 936"="" data-label="Page-Footer" href="https://developer.</p> </td> </tr> </tbody> </table> </td> </tr> </tbody> </table> </td> </tr> </tbody> </table> </div> <div data-bbox="> <p>Metodi</p> </p>
Nome	Type (Tipo)	Descrizione												
sequence	Array. <Object>	<p>La scorciatoia da tastiera per la registrazione.</p> <table border="1"> <thead> <tr> <th>Nome</th> <th>Type (Tipo)</th> <th>Descrizione</th> </tr> </thead> <tbody> <tr> <td>keyCode</td> <td>KeyboardEvent.keyCode</td> <td> <p>Il valore del tasto premuto dall'utente. Per ulteriori informazioni,,, consulta <a 102="" 55="" 920="" 936"="" data-label="Page-Footer" href="https://developer.</p> </td> </tr> </tbody> </table> </td> </tr> </tbody> </table> </td> </tr> </tbody> </table> </div> <div data-bbox="> <p>Metodi</p> </p></td></tr></tbody></table>	Nome	Type (Tipo)	Descrizione	keyCode	KeyboardEvent.keyCode	<p>Il valore del tasto premuto dall'utente. Per ulteriori informazioni,,, consulta <a 102="" 55="" 920="" 936"="" data-label="Page-Footer" href="https://developer.</p> </td> </tr> </tbody> </table> </td> </tr> </tbody> </table> </td> </tr> </tbody> </table> </div> <div data-bbox="> <p>Metodi</p> </p>						
Nome	Type (Tipo)	Descrizione												
keyCode	KeyboardEvent.keyCode	<p>Il valore del tasto premuto dall'utente. Per ulteriori informazioni,,, consulta <a 102="" 55="" 920="" 936"="" data-label="Page-Footer" href="https://developer.</p> </td> </tr> </tbody> </table> </td> </tr> </tbody> </table> </td> </tr> </tbody> </table> </div> <div data-bbox="> <p>Metodi</p> </p>												

Nome	Type (Tipo)	Descrizione		
		Nome	Type (Tipo)	Descrizione
				<p>mozilla.org/en-US/docs/Web/API/KeyboardEvent.key.</p>
				<p>LibraryE vt.positions chiavi da inviare. La posizione del tasto sulla tastiera. Per ulteriori informazioni,, consulta <a 102="" 55="" 920="" 936"="" data-label="Page-Footer" href="https://d eveloper.</p> </td> </tr> </tbody> </table> </div> <div data-bbox="> <p>Metodi</p> </p>

Nome	Type (Tipo)	Descrizione		
		Nome	Type (Tipo)	Descrizione
				Nome (Descrizione) mozilla.org/en-US/docs/Web/API/KeyboardEvent.location
		output	Array. <Object>	L'azione prevista da eseguire con la scorciatoia. Nome (Descrizione) KeyboardEvent.valueOf del tasto premuto dall'uten

Nome	Type (Tipo)	Descrizione		
		Nome	Type (Tipo)	Descrizione
				<p>Nome (nome) te. Per ulteriori informazioni,, consulta https://developer.mozilla.org/en-US/docs/Web/API/KeyboardEvent/key.</p> <p>1. keyCode (proprietà) - La posizione del tasto sulla</p>

Nome	Type (Tipo)	Descrizione		
		Nome	Type (Tipo)	Descrizione
				Tastiera. Per ulteriori informazioni,, consulta https://developer.mozilla.org/en-US/docs/Web/API/KeyboardEvent/location .

Valori restituiti:

Type (Tipo)

void

`requestDisplayConfig(highColorAccuracy) → {Promessa|Promessa. < {code: DisplayConfigErrorCode, messaggio: string} >}`

Richiede una configurazione di visualizzazione aggiornata dal server NICE DCV. Disponibile a partire da NICE DCV Web Client SDK 1.1.0 e NICE DCV Server 2022.0.

Parametri:

Nome	Type (Tipo)	Descrizione
<code>highColorAccuracy</code>	booleano	Se è necessario richiedere o meno un'elevata precisione del colore.

Valori restituiti:

Promessa. Se rifiutata, la promessa restituisce un oggetto di errore.

Type (Tipo)

Promessa | Promessa. < {code: [DisplayConfigErrorCode](#), messaggio: stringa} >

`requestDisplayLayout(layout) → {Promessa|Promessa. < {code: ResolutionErrorCode, messaggio: string} >}`

Richiede un layout di visualizzazione aggiornato per la connessione.

Parametri:

Nome	Type (Tipo)	Descrizione
<code>layout</code>	Array. < Monitora >	La richiesta viene visualizzata nel layout.

Valori restituiti:

Promessa. Se rifiutato, riceviamo un oggetto di errore.

Type (Tipo)

Promessa | Promessa. < {code: [ResolutionErrorCode](#), messaggio: stringa} >

RequestResolution (larghezza, altezza) → {Promessa | Promessa. < {code: [ResolutionErrorCode](#), messaggio: string} >}

Richiede una risoluzione dello schermo aggiornata dal server NICE DCV.

Parametri:

Nome	Type (Tipo)	Descrizione
width	numero	La larghezza da richiedere e in pixel. Il valore minimo consentito è 0.
height	numero	L'altezza da richiedere in pixel. Il valore minimo consentito è 0.

Valori restituiti:

Promessa. Se rifiutata, la promessa restituisce un oggetto di errore.

Type (Tipo)

Promessa | Promessa. < {code: [ResolutionErrorCode](#), messaggio: stringa} >

sendKeysEvent(evento) → {booleano}

Invia un evento scelta rapida da tastiera. Per ulteriori informazioni sugli eventi da tastiera, vedere <https://developer.mozilla.org/en-US/docs/Web/API/KeyboardEvent>. Gli eventi validi della tastiera includono: keydown, keypress, ekeyup. Per ulteriori informazioni su questi eventi, vedere [https://developer.mozilla.org/en-US/docs/Web/API/KeyboardEvent #events](https://developer.mozilla.org/en-US/docs/Web/API/KeyboardEvent#events).

Parametri:

Nome	Type (Tipo)	Descrizione
event	KeyboardEvent	L'evento da tastiera da inviare.

Valori restituiti:

Se l'evento non è valido, la funzione ritorna `false`. Se l'evento è valido, la funzione restituisce `true`.

Type (Tipo)

booleano

`sendKeyboardShortcut(scorciatoia) → {void}`

Consente di inviare la scelta rapida da tastiera. Utilizzate questa funzione per inviare un completo `keydown` o una `keyup` sequenza. Ad esempio, l'invio di `Ctrl + Alt + Del` invia gli `keydown` eventi per tutte le chiavi seguite dagli `keyup` eventi. Usa questa funzione anche se desideri inviare una sola chiave.

Parametri:

Nome	Type (Tipo)	Descrizione						
shortcut	Array. <Object>	L'array di chiavi da inviare. <table border="1" data-bbox="1068 1409 1511 1879"> <thead> <tr> <th>Nome</th> <th>Type (Tipo)</th> <th>Descrizione</th> </tr> </thead> <tbody> <tr> <td>key</td> <td>KeyboardEvent.chiave</td> <td>Il valore del tasto premuto dall'utente. Per</td> </tr> </tbody> </table>	Nome	Type (Tipo)	Descrizione	key	KeyboardEvent.chiave	Il valore del tasto premuto dall'utente. Per
Nome	Type (Tipo)	Descrizione						
key	KeyboardEvent.chiave	Il valore del tasto premuto dall'utente. Per						

Nome	Type (Tipo)	Descrizione		
		Nome	Type (Tipo)	Descrizione
				ulteriori informazioni,, consulta https://developer.mozilla.org/en-US/docs/Web/API/KeyboardEvent/key .
		location	KeyboardEvent.position	L'array di chiavi da inviare. La posizione del tasto sulla tastiera. Per ulteriori informazioni,,

Nome	Type (Tipo)	Descrizione		
		Nome	Type (Tipo)	Descrizione
				consulta https:// d eveloper. mozilla.o rg/en- US/ docs/ Web/ API/ Keybo ardEvent / location .

Valori restituiti:

Type (Tipo)

void

`setDisplayQuality(min, max opt) → {void}`

Imposta la qualità dell'immagine da utilizzare per la connessione. L'intervallo valido è 0 pari a 100, con 1 la qualità dell'immagine più bassa e 100 la qualità dell'immagine più alta. Specifica 0 per mantenere il valore corrente.

Parametri:

Nome	Type (Tipo)	Attributi	Descrizione
min	numero		La qualità minima dell'immagine.
max	numero	<optional>	La massima qualità dell'immagine.

Valori restituiti:

Type (Tipo)

void

setDisplayScale(ScaleRatio, DisplayID) → {Promessa | Promessa. < {code: [ResolutionErrorCode](#), message: string} >} (OBSOLETO)

Obsoleto dalla versione 1.3.0. Non è più necessario impostare la scala di visualizzazione. Le coordinate del mouse verranno gestite automaticamente internamente.

Notifica al NICE DCV che il display è ridimensionato sul lato client. Usalo per notificare al server che deve ridimensionare gli eventi del mouse in modo che corrispondano al rapporto di visualizzazione del client.

Parametri:

Nome	Type (Tipo)	Descrizione
scaleRatio	float	Il rapporto di scala da utilizzare. Deve essere un numero strettamente positivo.
displayId	numero	L'ID del display da ridimensionare.

Valori restituiti:

Promessa. Se rifiutata, la promessa restituisce un oggetto di errore.

Type (Tipo)

Promessa | Promessa. < {code: [ResolutionErrorCode](#), messaggio: stringa} >

setKeyboardQuirks(stranezze) → {void}

Imposta le stranezze della tastiera per il computer client.

Parametri:

Nome	Type (Tipo)	Descrizione						
quirks	Oggetto	La tastiera si comporta in modo strano per abilitare o disabilitare.						
		<table border="1"> <thead> <tr> <th>Nome</th> <th>Type (Tipo)</th> <th>Descrizione</th> </tr> </thead> <tbody> <tr> <td>macOptiToAlt</td> <td>booleanc</td> <td>Per mappare il tasto Opzione su Alt per macOS, specifica true. In caso contrario, specifica false.</td> </tr> </tbody> </table>	Nome	Type (Tipo)	Descrizione	macOptiToAlt	booleanc	Per mappare il tasto Opzione su Alt per macOS, specifica true. In caso contrario, specifica false.
Nome	Type (Tipo)	Descrizione						
macOptiToAlt	booleanc	Per mappare il tasto Opzione su Alt per macOS, specifica true. In caso contrario, specifica false.						

Nome	Type (Tipo)	Descrizione		
		Nome	Type (Tipo)	Descrizione
		macCommandToControl	booleano	Per mappare il tasto Comando su Ctrl per macOS, specifica true. In caso contrario, specifica false.

Valori restituiti:

Type (Tipo)

void

setMaxDisplayRisoluzione (larghezza massima, altezza massima) → {void}

Imposta la risoluzione massima dello schermo da utilizzare per la connessione.

Parametri:

Nome	Type (Tipo)	Descrizione
<code>maxWidth</code>	numero	Larghezza massima dello schermo in pixel. Il valore minimo consentito è 0.
<code>maxHeight</code>	numero	L'altezza massima dello schermo in pixel. Il valore minimo consentito è 0.

Valori restituiti:

Type (Tipo)

`void`

`setMicrophone (abilita) → {Promessa|Promessa. < {code: AudioErrorCode, messaggio: string} >}`

Consente di attivare o disattivare il microfono.

Parametri:

Nome	Type (Tipo)	Descrizione
<code>enable</code>	booleano	Per abilitare il microfono, specificare <code>true</code> . Per disattivare il microfono, specifica <code>false</code> .

Valori restituiti:

Promessa. Se rifiutata, la promessa restituisce un oggetto di errore.

Type (Tipo)

Promessa | Promessa. < {code: [AudioErrorCode](#), messaggio: stringa} >

setMinDisplayRisoluzione (larghezza minima, altezza minima) → {void}

Imposta la risoluzione minima dello schermo da utilizzare per la connessione. Alcune applicazioni potrebbero richiedere una risoluzione minima dello schermo. Se la risoluzione minima richiesta è maggiore della risoluzione massima supportata dal client, viene utilizzata una strategia di ridimensionamento. Usa questa funzione con attenzione. La strategia di ridimensionamento potrebbe causare un sistema di input tattile e mouse meno preciso.

Parametri:

Nome	Type (Tipo)	Descrizione
minWidth	numero	La larghezza minima dello schermo in pixel. Il valore minimo consentito è 0.
minHeight	numero	L'altezza minima dello schermo in pixel. Il valore minimo consentito è 0.

Valori restituiti:

Type (Tipo)

void

setUploadBandwidth(valore) → {numero}

Imposta la larghezza di banda massima da utilizzare per caricare i file sul server NICE DCV.

Parametri:

Nome	Type (Tipo)	Descrizione
value	numero	Il limite massimo di larghezza di banda in kbps. L'intervallo valido è compreso tra 1024 kbps e 102400 kbps.

Valori restituiti:

- Il limite di larghezza di banda impostato. null se la funzionalità di archiviazione dei file è disattivata sul server.

Type (Tipo)

numero

setVolume (volume) → {void}

Imposta il livello del volume da utilizzare per l'audio. L'intervallo valido è compreso tra 0 e 100, dove 0 è il volume più basso e 100 il volume più alto.

Parametri:

Nome	Type (Tipo)	Descrizione
volume	numero	Il livello del volume da utilizzare.

Valori restituiti:

Type (Tipo)

void

`setWebcam (abilita, ID dispositivo) → {Promessa | Promessa. < {code: WebcamErrorCode, messaggio: string} >}`

Consente di attivare o disattivare la webcam.

Parametri:

Nome	Type (Tipo)	Descrizione
<code>enable</code>	booleano	Per abilitare la webcam, specificare <code>true</code> . Per disattivare la webcam, specifica <code>false</code> .
<code>deviceId</code>	string	L'ID del dispositivo della webcam.

Valori restituiti:

Prometti che, in caso di successo, si risolve nel `DeviceID` della webcam allegata/scollegata. Se rifiutata, la promessa restituisce un oggetto di errore.

Type (Tipo)

Promessa | Promessa. < {code: [WebcamErrorCode](#), messaggio: stringa} >

`syncClipBoards () → {booleano}`

Sincronizza gli appunti del client locale con gli appunti del server NICE DCV remoto. La copia automatica deve essere supportata dal browser.

Valori restituiti:

Se gli appunti sono stati sincronizzati, la funzione ritorna `true`. Se gli appunti non sono stati sincronizzati o se il browser non supporta la copia automatica, la funzione ritorna `false`.

Type (Tipo)

booleano

Classe di autenticazione

La classe di autenticazione deve essere utilizzata per ottenere un token di autenticazione chiamando il [authenticatemetodo](#) del `dcv` modulo. Per un esempio che mostra come usarlo, consulta la sezione [Guida introduttiva](#).

Esponde

- [Metodi](#)

Metodi

Elenco

- [riprova \(\) → {void}](#)
- [SendCredenziali \(credenziali\) → {void}](#)

`riprova () → {void}`

Riprova il processo di autenticazione.

Valori restituiti:

Type (Tipo)

`void`

`SendCredenziali (credenziali) → {void}`

Invia le credenziali di autenticazione fornite dal client al server NICE DCV.

Parametri:

Nome	Type (Tipo)	Descrizione
<code>credentials</code>	Oggetto	L'oggetto contenente le credenziali fornite. Le credenziali devono avere lo stesso nome ed essere dello

Nome	Type (Tipo)	Descrizione
		stesso tipo specificato nella sfida.

Valori restituiti:

Type (Tipo)

void

Categoria risorsa

La classe di risorse può recuperare o eliminare il file corrispondente appena stampato o scaricato. Quando si eseguono queste azioni, l'osservatore corrispondente funziona [filePrintede](#) [fileDownload](#)verrebbe rispettivamente richiamato con l'oggetto risorsa come unico argomento. Tale risorsa può essere accettata o rifiutata per recuperare o scartare il file a cui fa riferimento.

Espone

- [Metodi](#)

Metodi

Elenco

- [accept \(parametri URL\) → {void}](#)
- [declino \(\) → {void}](#)

[accept \(parametri URL\) → {void}](#)

Scarica localmente la risorsa.

Parametri:

Nome	Type (Tipo)	Descrizione
<code>urlParameters</code>	Oggetto	L'oggetto opzionale contenent e le coppie chiave/valore dei parametri di ricerca URL passati alla richiesta di recupero della risorsa.

Valori restituiti:

Type (Tipo)

`void`

`declino () → {void}`

Scarta la risorsa.

Valori restituiti:

Type (Tipo)

`void`

SDK dell'interfaccia utente Web NICE DCV

Una libreria di componenti JavaScript React, che attualmente esporta un singolo componente React chiamato `DCVViewer` che si connette al server NICE DCV e rende la barra degli strumenti per interagire con lo stream remoto.

Esponde

- [Componenti](#)

Componenti

Elenco

- [Visualizzatore DCV](#)

Visualizzatore DCV

Il componente React che rende la barra degli strumenti con tutte le sue funzionalità utili per interagire con lo stream remoto.

Proprietà:

Elenco

- [dcv](#)
- [UIConfig](#)

dcv

Nome	Type (Tipo)	Campo obbligatorio	Descrizione
dcv	Oggetto	Si	L'oggetto che definisce le proprietà necessarie per stabilire la connessione al server NICE DCV, impostando il livello di registro e l'URL da cui caricare le risorse SDK del NICE DCV Web Client e accedere alle risorse DCV.

Nome	Type (Tipo)	Campo obbligatorio	Descrizione																								
			<table border="1"> <thead> <tr> <th>Non</th> <th>Type</th> <th>Car</th> <th>Descrizio</th> </tr> <tr> <th></th> <th>(Tip</th> <th>obbl</th> <th>ne</th> </tr> <tr> <th></th> <th>o)</th> <th>rio</th> <th></th> </tr> </thead> <tbody> <tr> <td>ses</td> <td>Strir</td> <td>Sì</td> <td>L'ID della sessione NICE DCV.</td> </tr> <tr> <td>aut</td> <td>Strir</td> <td>Sì</td> <td>Il token di autenticazione da utilizzare per la connessione al server.</td> </tr> <tr> <td>ser</td> <td>Strir</td> <td>Sì</td> <td>Il nome host e la porta del server NICE DCV</td> </tr> </tbody> </table>	Non	Type	Car	Descrizio		(Tip	obbl	ne		o)	rio		ses	Strir	Sì	L'ID della sessione NICE DCV.	aut	Strir	Sì	Il token di autenticazione da utilizzare per la connessione al server.	ser	Strir	Sì	Il nome host e la porta del server NICE DCV
Non	Type	Car	Descrizio																								
	(Tip	obbl	ne																								
	o)	rio																									
ses	Strir	Sì	L'ID della sessione NICE DCV.																								
aut	Strir	Sì	Il token di autenticazione da utilizzare per la connessione al server.																								
ser	Strir	Sì	Il nome host e la porta del server NICE DCV																								

Nome	Type (Tipo)	Campo obbligatorio	Descrizione			
			Non	Type (Tipo)	Carri obbligatorio	Descrizione
						<p>in esecuzione nel seguente formato: https:// d cv_host_address:port. Ad esempio: https://:8443- dcv-server.</p>
			base	String	Sì	L'URL assoluto o relativo da cui caricare i file SDK.
			base	String	No (imp)	L'URL assoluto

Nome	Type (Tipo)	Campo obbligatorio	Descrizione			
			Non	Type (Tipo)	Campo obbligatorio	Descrizione
						ione o pred relativo ta:" da cui accedere alle risorse DCV.
			onD ect	funz	No (imp ione di pred ta:(=> {})	La funzione di callback viene richiamat a durante la disconnes sione dal server NICE DCV e la connessio ne viene chiusa.

Nome	Type (Tipo)	Campo obbligatorio	Descrizione																
			<table border="1"> <thead> <tr> <th>Non</th> <th>Type</th> <th>Carri</th> <th>Descrizio</th> </tr> <tr> <th></th> <th>(Tip</th> <th>obbl</th> <th>ne</th> </tr> <tr> <th></th> <th>rio</th> <th></th> <th></th> </tr> </thead> <tbody> <tr> <td>log</td> <td>Log</td> <td>No</td> <td>Il (imp livello ione di pred registro ta:Lo da vel. INF utilizzar e nel visualizz atore.</td> </tr> </tbody> </table>	Non	Type	Carri	Descrizio		(Tip	obbl	ne		rio			log	Log	No	Il (imp livello ione di pred registro ta:Lo da vel. INF utilizzar e nel visualizz atore.
Non	Type	Carri	Descrizio																
	(Tip	obbl	ne																
	rio																		
log	Log	No	Il (imp livello ione di pred registro ta:Lo da vel. INF utilizzar e nel visualizz atore.																

UIConfig

Nome	Type (Tipo)	Campo obbligatorio	Descrizione
uiConfig	Oggetto	No (impostazione predefinita: {})	L'oggetto che definisce le proprietà per configurare se la barra degli strumenti è visibile e se visualizzare i pulsanti a schermo intero e multimonitor su di essa.

Nome	Type (Tipo)	Campo obbligatorio	Descrizione								
			<table border="1"> <thead> <tr> <th data-bbox="1169 224 1268 401">Nome</th> <th data-bbox="1268 224 1349 401">Type (Tipo)</th> <th data-bbox="1349 224 1430 401">Campo obbligatorio</th> <th data-bbox="1430 224 1620 401">Descrizione</th> </tr> </thead> <tbody> <tr> <td data-bbox="1169 401 1268 1850">too</td> <td data-bbox="1268 401 1349 1850">Ogg</td> <td data-bbox="1349 401 1430 1850">No</td> <td data-bbox="1430 401 1620 1850">L'oggetto (imp che ione definisce pred le ta: { opzioni di configura zione per la barra degli strumenti .</td> </tr> </tbody> </table>	Nome	Type (Tipo)	Campo obbligatorio	Descrizione	too	Ogg	No	L'oggetto (imp che ione definisce pred le ta: { opzioni di configura zione per la barra degli strumenti .
Nome	Type (Tipo)	Campo obbligatorio	Descrizione								
too	Ogg	No	L'oggetto (imp che ione definisce pred le ta: { opzioni di configura zione per la barra degli strumenti .								

Nome	Type (Tipo)	Campo obbligatorio	Descrizione			
			Non	Type (Tipo)	Campo obbligatorio	Descrizione
						Descrizione (campo obbligatorio)
						strumento.
						<p>Descrizione (predefinito) definire predefinito (strumento) o nascondere il pulsante a schermo intero sulla barra degli strumenti.</p>
						<p>Descrizione (predefinito) definire predefinito (strumento) o</p>

Nome	Type (Tipo)	Campo obbligatorio	Descrizione				
			Non	Type (Tipo)	Carri obbl rio	Descrizio ne	Descrizione obbligato rio nasconde e il pulsante multimon tor sulla barra degli strument .

Note di rilascio e cronologia dei documenti per NICE DCV Web Client SDK

Questa pagina fornisce le note di rilascio e la cronologia dei documenti per NICE DCV Web Client SDK.

Argomenti

- [Note di rilascio di NICE DCV Web Client SDK](#)
- [Cronologia dei documenti](#)

Note di rilascio di NICE DCV Web Client SDK

Questa sezione fornisce le note di rilascio per l'SDK NICE DCV Web Client in base alla data di rilascio.

Argomenti

- [1.5.10 — 19 dicembre 2023](#)
- [1.5.6 — 9 novembre 2023](#)
- [1.4.4 — 29 giugno 2023](#)
- [1.4.0 — 28 marzo 2023](#)
- [1.3.1 — 9 dicembre 2022](#)
- [1.3.0 — 11 novembre 2022](#)
- [1.2.1 — 21 luglio 2022](#)
- [1.2.0 — 29 giugno 2022](#)
- [1.1.3 — 23 maggio 2022](#)
- [1.1.2 — 19 maggio 2022](#)
- [1.1.1 — 23 marzo 2022](#)
- [1.1.0 — 23 febbraio 2022](#)
- [1.0.4 — 20 dicembre 2021](#)
- [1.0.3 — 01 settembre 2021](#)
- [1.0.2 — 30 luglio 2021](#)

- [1.0.1 — 31 maggio 2021](#)
- [1.0.0 — 24 marzo 2021](#)

1.5.10 — 19 dicembre 2023

Versione	Note di rilascio
<ul style="list-style-type: none"> • Versione semantica: 1.5.10 • Costruzione: 684 	<p>Modifiche e correzioni di bug</p> <ul style="list-style-type: none"> • Correggi gli errori di decodifica dello stream

1.5.6 — 9 novembre 2023

Versione	Note di rilascio
<ul style="list-style-type: none"> • Versione semantica: 1.5.6 • Costruzione: 659 	<p>Modifiche e correzioni di bug</p> <ul style="list-style-type: none"> • Miglioramenti delle prestazioni nella decodifica e nel rendering degli stream • Supporto rimosso per Internet Explorer 11

1.4.4 — 29 giugno 2023

Versione	Note di rilascio
<ul style="list-style-type: none"> • Versione semantica: 1.4.4 • Build: 573 	<p>Modifiche e correzioni di bug</p> <ul style="list-style-type: none"> • Il componente dell'interfaccia utente del visualizzatore ora utilizza l'<code>navigator.keyboard.lock</code> API sui browser che

Versione	Note di rilascio
	<p>lo supportano per gestire chiavi speciali a schermo intero.</p> <ul style="list-style-type: none"> • È stato risolto un problema che poteva causare colori errati quando si utilizzava Chrome 114 o versioni successive. • WebCodecs Rilevamento migliorato. • Risolto un problema relativo allo stato del pulsante del mouse quando si accedeva alla finestra. • È stato risolto un problema che poteva far sì che i tasti modificatori rimanessero premuti su macOS. • Maggiore robustezza audio in condizioni di rete degradate. • Risolte le perdite di memoria. • Registri migliorati per includere ora e livello.

1.4.0 — 28 marzo 2023

Versione	Note di rilascio
<ul style="list-style-type: none"> • Versione semantica: 1.4.0 • Costruzione: 476 	<p>Nuove funzionalità</p> <ul style="list-style-type: none"> • Aggiunto un nuovo <code>uploadFiles</code> metodo all'<code>FileStorage</code> oggetto per caricare più file. •

Versione	Note di rilascio
	<p>Il componente dell'interfaccia utente del visualizzatore ora supporta il drag and drop per avviare il caricamento dei file.</p> <ul style="list-style-type: none">• L'API WebCodecs del browser viene ora utilizzata anche per l'audio e la webcam. <p>Modifiche e correzioni di bug</p> <ul style="list-style-type: none">• Sono state corrette le perdite di memoria relative alle connessioni ripetute dalla stessa pagina.• <code>setUploadBandwidth</code> ora consente valori fino a 1 Gbps.• Rendering ottimizzato dei componenti dell'interfaccia utente.• Supporto fisso per cursori animati su Windows.• È stato risolto un problema relativo al supporto degli appunti quando per la stessa operazione sono presenti sia dati di testo che dati di immagine.• Maggiore robustezza dell'API Webcam: le impostazioni non possono essere modificate e mentre una richiesta è già in corso, <code>webcam.setEnabled</code> ora tiene traccia dell'ID del dispositivo per la richiesta in corso e restituisce una Promise. Il component

Versione	Note di rilascio
	e dell'interfaccia utente del visualizzatore mostra una notifica in caso di errore.

1.3.1 — 9 dicembre 2022

Versione	Note di rilascio
<ul style="list-style-type: none">• Versione semantica: 1.3.1• Costruzione: 413	<p>Modifiche e correzioni di bug</p> <ul style="list-style-type: none">• È stato risolto un problema che poteva causare la mancata sincronizzazione dell'interfaccia utente di reindirizzamento del fuso orario con il server.• È stata risolta una perdita di memoria dopo più riconessioni.• È stato risolto un problema che causava una pagina vuota durante la disconnessione.• Risolto un bug che causava la chiusura degli avvisi della console sul decoder audio.

1.3.0 — 11 novembre 2022

Versione	Note di rilascio
<ul style="list-style-type: none">• Versione semantica: 1.3.0• Compilazione: 407	<p>Nuove funzionalità</p> <ul style="list-style-type: none">• Ha adottato Cloudscape (https://cloudscape.design) per il componente UI Viewer.

Versione	Note di rilascio
	<ul style="list-style-type: none">È stato aggiunto il supporto per il reindirizzamento del fuso orario. <p>Modifiche e correzioni di bug</p> <ul style="list-style-type: none">Risolto il problema dell'aggiornamento mancante sugli appunti asincroni quando il visualizzatore DCV è attivo.La <code>setDisplayScale</code> funzione non è più necessaria quando si ridimensiona lo schermo sul lato client.Il <code>DCVViewer</code> componente ora chiama automaticamente <code>disconnect()</code> quando viene smontato.

1.2.1 — 21 luglio 2022

Versione	Note di rilascio
<ul style="list-style-type: none">Versione semantica: 1.2.1Costruzione: 358	<p>Modifiche e correzioni di bug</p> <ul style="list-style-type: none">È stato risolto un problema che causava l'errore di connessione al server NICE DCV 2019.1 e versioni precedenti.

1.2.0 — 29 giugno 2022

Versione	Note di rilascio
<ul style="list-style-type: none">• Versione semantica: 1.2.0• Costruzione: 352	<p>Modifiche e correzioni di bug</p> <ul style="list-style-type: none">• Risolto un bug che si bloccava quando i frame ricevuti erano più grandi della risoluzione massima supportata (4096x2160).• Gli oggetti di risorsa (passati come argomenti <code>fileDownload</code> e <code>filePrinted</code> osservatori) ora dispongono dei <code>decline</code> metodi <code>accept</code> e che possono essere chiamati sull'oggetto rispettivamente per scaricare e scartare la risorsa.• Correzione di un bug minore relativo alla sincronizzazione automatica degli appunti durante la disconnessione.

1.1.3 — 23 maggio 2022

Versione	Note di rilascio
<ul style="list-style-type: none">• Versione semantica: 1.1.3• Costruzione: 329	<p>Modifiche e correzioni di bug</p> <ul style="list-style-type: none">• È stato risolto un problema che impediva la corretta connessione quando si specificava l' <code>web-url-path</code> opzione.

1.1.2 — 19 maggio 2022

Versione	Note di rilascio
<ul style="list-style-type: none">• Versione semantica: 1.1.2• Costruzione: 322	<p>Modifiche e correzioni di bug</p> <ul style="list-style-type: none">• È stato risolto un problema che poteva causare il malfunzionamento dell'input dopo la connessione.• Coordinate fisse del mouse quando il rapporto di scala è maggiore di 1.

1.1.1 — 23 marzo 2022

Versione	Note di rilascio
<ul style="list-style-type: none">• Versione semantica: 1.1.1• Build: 309	<p>Modifiche e correzioni di bug</p> <ul style="list-style-type: none">• Segnala <code>Transport Error</code> il timeout della comunicazione con il server.• È stato corretto un errore di decodifica ricorrente durante lo streaming di grandi risoluzioni.

1.1.0 — 23 febbraio 2022

Versione	Note di rilascio
<ul style="list-style-type: none">• Versione semantica: 1.1.0	<p>Nuove funzionalità</p>

Versione	Note di rilascio
<ul style="list-style-type: none"> Build: 295 	<ul style="list-style-type: none"> Rilascia la libreria SDK NICE DCV Web UI con <code>DCVViewer</code> il componente React. Esporta NICE DCV Web Client SDK sia come moduli UMD che ES. Aggiunto il supporto per un'elevata precisione del colore. È stata aggiunta la possibilità di elencare e interagire con i client connessi a una sessione. Aggiunte notifiche per la connessione e la disconnessione. <p>Modifiche e correzioni di bug</p> <ul style="list-style-type: none"> Supporto migliorato per la decodifica dei webcodecs. Vari miglioramenti alla tastiera. Risolto un bug che impediva l'apertura di una seconda schermata quando gli appunti erano disabilitati.

1.0.4 — 20 dicembre 2021

Versione	Note di rilascio
<ul style="list-style-type: none"> Versione semantica: 1.0.4 	<p>Nuove funzionalità</p> <ul style="list-style-type: none">

Versione	Note di rilascio
Costruzione: 249	<p>Supporta l'apertura di più connessioni dalla stessa pagina.</p> <ul style="list-style-type: none">• Supporta il caricamento dell'SDK da un CDN.

1.0.3 — 01 settembre 2021

Versione	Note di rilascio
<ul style="list-style-type: none">• Versione semantica: 1.0.3• Costruzione: 202	<p>Nuove funzionalità</p> <ul style="list-style-type: none">• Supporto sperimentale per WebCodecs . Questo è disabilitato per impostazione predefinita e deve essere abilitato tramite l'<code>ConnectionConfig</code> oggetto utilizzando la nuova proprietà <code>enableWebCodecs</code> .• Appunti: è stato aggiunto il supporto per il tipo di <code>image/png</code> dati nei browser basati su Chromium.• Aggiunto <code>observer/callback</code> per ottenere lo screenshot del server come immagine PNG (richiede il server NICE DCV 2021.2). <p>Modifiche e correzioni di bug</p> <ul style="list-style-type: none">• Migliore gestione dei modificatori di tastiera.

1.0.2 — 30 luglio 2021

Versione	Note di rilascio
<ul style="list-style-type: none"> • Versione semantica: 1.0.2 • Costruzione: 167 	<ul style="list-style-type: none"> • Rilevamento della pressione fissa per eventi legati allo stilo. • Supporto migliorato per il layout di tastiera coreano su Chrome.

1.0.1 — 31 maggio 2021

Versione	Note di rilascio
<ul style="list-style-type: none"> • Versione semantica: 1.0.1 • Build: 141 	<ul style="list-style-type: none"> • Risolto il problema della propagazione degli errori di connessione e dei relativi motivi • È stato corretto l'aggiornamento dello stato di avanzamento dei blocchi di archiviazione dei file • Migliore gestione della webcam • Elaborazione dell'ingresso audio migliorata

1.0.0 — 24 marzo 2021

Versione	Note di rilascio
<ul style="list-style-type: none"> • Versione semantica: 1.0.0 • 	<p>Versione iniziale del NICE DCV Web Client SDK.</p>

Versione	Note di rilascio
Compilazione: 81	

Cronologia dei documenti

La tabella seguente descrive la documentazione per questa versione di NICE DCV Web Client SDK.

Modifica	Description	Data
NICE DCV Web Client SDK versione 1.5.6	NICE DCV Web Client SDK 1.5.6 è ora disponibile. Per ulteriori informazioni, consultate SDK v.1.5.6.	9 novembre 2023
NICE DCV Web Client SDK versione 1.4.4	NICE DCV Web Client SDK 1.4.4 è ora disponibile. Per ulteriori informazioni, consultate SDK v.1.4.4.	29 giugno 2023
NICE DCV Web Client SDK versione 1.4.0	NICE DCV Web Client SDK 1.4.0 è ora disponibile. Per ulteriori informazioni, vedete SDK v.1.4.0.	28 marzo 2023
NICE DCV Web Client SDK versione 1.3.1	NICE DCV Web Client SDK 1.3.1 è ora disponibile. Per ulteriori informazioni, consultate SDK v.1.3.1.	9 dicembre 2022
NICE DCV Web Client SDK versione 1.3.0	NICE DCV Web Client SDK 1.3.0 è ora disponibile. Per ulteriori informazioni, vedete SDK v.1.3.0.	11 novembre 2022

Modifica	Description	Data
NICE DCV Web Client SDK versione 1.2.0	NICE DCV Web Client SDK 1.2.0 è ora disponibile. Per ulteriori informazioni, vedete SDK v.1.2.0.	29 giugno 2022
NICE DCV Web Client SDK versione 1.1.0	NICE DCV Web Client SDK 1.1.0 è ora disponibile. Per ulteriori informazioni, vedete SDK v.1.1.0.	23 febbraio 2022
NICE DCV Web Client SDK versione 1.0.1	Corretti alcuni errori di battitura. Risolti alcuni bug minori, vedi SDK v.1.0.1.	31 maggio 2021
Rilascio iniziale	Prima pubblicazione di questi contenuti.	24 marzo 2021

Le traduzioni sono generate tramite traduzione automatica. In caso di conflitto tra il contenuto di una traduzione e la versione originale in Inglese, quest'ultima prevarrà.