

DEVELOPER API

1

Senape.tv

Piattaforma per la distribuzione di streaming live e on-demand

Documentazione tecnica

Developer API

Versione 2.0.0

MANUALI OPERATIVI

Documentazione tecnica – Developer API

Sommario

INTRODUZIONE	iii
CAPITOLO 1	
Concetti di base	1
I widget	3
Schede tecniche dei widget	5
Widget Archivio	5
Widget Ricerca	7
CAPITOLO 2	
Le API	8
Ricerca e selezione video	8
Reperimento dettagli video	10
Reperimento dettagli categorie	11
Reperimento dettagli programmazione canale	12
Accesso diretto all'immagine associata al singolo filmato/live	13
CAPITOLO 3	
Web Service	14
CAPITOLO 4	
Il Pushing	16
Funzionamento	16
Struttura dei dati	17
Inizializzazione della base dati e successive sincronizzazioni	22
CAPITOLO 5	
L'importazione "trasparente" dei video	24
CAPITOLO 6	
Le statistiche	27

REQUISITI E SPECIFICHE DI SISTEMA

CAPITOLO 7

Modalità accesso sicuro 28

APPENDICE

Utilizzo dell'Encoder Flash Media Live 30

Trasmissione di una diretta radio 33



Senape.tv è la piattaforma italiana di riferimento per lo streaming video professionale

Un innovativo sistema di distribuzione multicanale dei contenuti live e on-demand perfettamente integrabile su portali esterni e applicazioni di terze parti.

Lo streaming è il sistema più veloce per la distribuzione dei contenuti video su internet poiché invia al player dell'utente solamente i frame da riprodurre in un dato momento, senza dover prima scaricare per intero il filmato sul computer o smartphone. I dati vengono ricevuti poco alla volta e riprodotti pochi secondi dopo l'inizio della ricezione. Per questo stesso motivo è anche il sistema più sicuro per tutelare un filmato dalle copie indesiderate dal momento che nessun dato viene salvato nella cache del browser dell'utente.

Lo streaming, inoltre, è l'unica modalità di trasmissione dati in grado di consentire la distribuzione di contenuti "in diretta".

Senape.tv è una piattaforma tecnologica estremamente versatile che consente di allestire in tempi rapidi portali dedicati all'erogazione in streaming di contenuti multimediali fruibili via web, permettendo una distribuzione controllata e ramificata dei contenuti verso gli opportuni soggetti interessati (clienti, collaboratori, competitor, semplici visitatori occasionali, ...).

Senape.tv si spinge oltre la consueta collezione di filmati, offrendo nuove funzionalità uniche nel loro genere e garantendo una gestione semplificata, flessibile, e particolarmente fine nella creazione di palinsesti e archivi tematici, nonché di dirette interattive, senza necessità di investire in costose infrastrutture tecnologiche (nessun hardware da acquistare, nessuna

REQUISITI E SPECIFICHE DI SISTEMA

integrazione con software preesistente) e senza bisogno di possedere competenze specifiche nell'ambito della programmazione.

Grazie ai *widget* di Senape.tv, arricchire il proprio sito o una intranet aziendale si riduce ad un semplice copia-e-incolla di poche righe di codice, in quanto tutte le funzionalità messe a disposizione dalla piattaforma sono *component-oriented*.

Gli sviluppatori, poi, possono integrare le funzionalità di video streaming messe a disposizione da Senape.tv all'interno delle proprie applicazioni mediante le API (*Application Programmer Interface*) esposte dalla piattaforma.

La piattaforma Senape.tv ha una interfaccia interamente web e può essere acceduta ovunque – tramite un semplice browser – previa autenticazione con le proprie credenziali all'URL <https://stream.senape.tv>.

Concetti di base

Dirette, video, playlist e canali.

I concetti fondanti di Senape.tv, i mattoni di base per costruire molto di più.

La piattaforma Senape.tv vede come suo elemento fondamentale il **video** che viene distribuito a tutti gli utenti che lo richiedono.

Ogni video caricato dall'utente può essere distribuito su internet in streaming grazie ad un serie di operazioni che vengono effettuate automaticamente dalla piattaforma.

Una volta caricato (in uno qualsiasi dei formati accettati), un video viene *convertito in un formato adatto alla distribuzione sul web*, riadattato nelle dimensioni e nella qualità al fine di poter raggiungere il più ampio numero di dispositivi (pc, tablet, smartphone, ecc.).

Il caricamento dei video su Senape.tv può essere effettuato direttamente tramite browser per video di dimensioni ridotte (fino a 100 MB), mentre per file video di dimensioni superiori occorre utilizzare l'apposita app **SenapeUpload** – scaricabile gratuitamente seguendo il link presente nella sezione *Strumenti* della pagina principale della piattaforma.

Terminata la conversione la piattaforma notifica l'utente dell'awenuto completamento e dell'effettiva disponibilità del video che – da quel momento in poi – può essere fruito in modalità on-demand.

Assieme ai video la piattaforma è in grado di erogare anche **dirette**, ovvero contenuti live.

Per la trasmissione di una diretta occorre disporre di uno strumento di cattura video (es. una telecamera, una webcam, ecc.) e di un programma specifico di codifica del flusso dati chiamato **encoder**.

REQUISITI E SPECIFICHE DI SISTEMA

L'encoder – necessario per la pubblicazione di una diretta su Senape.tv – richiede il supporto del *protocollo RTMP* pertanto si consiglia l'utilizzo di [Adobe Flash Media Live Encoder](#), software gratuito che può essere scaricato direttamente dal sito di Adobe (attualmente disponibile nella versione 3.2 per Windows e per Mac). In alternativa esistono soluzioni commerciali a pagamento – che si arricchiscono di funzionalità più evolute come, ad esempio, una regia software o la possibilità di aggiungere titoli in sovrapposizione al flusso video – quali [Wirecast](#) (attualmente disponibile nella versione 4.2 per Windows e per Mac).

Durante la trasmissione di una diretta la piattaforma Senape.tv consente di accedere ad una **cabina di regia** web (raggiungibile dalla pagina di dettaglio dell'evento live). Dalla cabina di regia è possibile visualizzare ciò che si sta trasmettendo in quel momento, come pure avviare e interrompere la registrazione dell'evento stesso sul cloud di Senape.tv.

La registrazione sul cloud si rivela preziosa qualora si desideri rendere disponibile il filmato della trasmissione non appena l'evento termina: il video trasmesso potrà poi essere aggiornato successivamente – e quindi sostituito – con una versione rimontata dell'evento (ad esempio per rimuovere i “tempi morti” o comunque per migliorare la qualità del contenuto stesso).

I video possono anche essere organizzati in playlist.

Una **playlist** può essere definita secondo vari criteri: ad esempio, può contenere una lista specifica di filmati, piuttosto che una sequenza generata automaticamente utilizzando gli ultimi video pubblicati, archiviati o promossi come “in evidenza” o, ancora, appartenenti ad una o più categorie tematiche.

Video, live e playlist, infine, sono elementi che possono essere presentati in un canale.

Il **canale** è un componente della piattaforma in grado di modificare il proprio contenuto in relazione ad una pianificazione temporale delle trasmissioni. Ad esempio, si può definire che in un certo canale dalle ore 8:00 alle 10:00 vada in onda una diretta, mentre dalle 10:00 alla mezzanotte venga visualizzata una playlist in rotazione degli ultimi 10 video caricati. I canali di Senape.tv concettualmente si avvicinano molto (pur non eguagliandoli) ai tradizionali canali televisivi guidati da un palinsesto.

I widget

*Incorporare i tuoi contenuti su siti, portali e app è facile.
Basta un copia e incolla.*

La piattaforma Senape.tv è prima di tutto un'area "amministrativa" dei propri contenuti, dunque non pubblicamente accessibile al pubblico. La fruizione pubblica dei video e degli altri elementi della piattaforma si affida all'utilizzo di semplici codici (<script> o <iframe>), chiamati **widget**, che possono essere utilizzati all'interno di un qualunque sito web: la piattaforma è dunque un semplice strumento che consente di arricchire uno o più siti dell'utente con contenuti video multimediali erogati in streaming.

In alternativa un utente può utilizzare la propria **pagina personale**, ospitata sulla piattaforma Senape.tv, per la pubblicazione e la condivisione dei propri contenuti (landing page dell'utente).

Ogni elemento della piattaforma può essere facilmente condiviso attraverso l'apposito tasto "Condividi", attraverso il quale è possibile copiare e personalizzare lo script da utilizzare.

I widget disponibili sono i seguenti:

- **Widget Video**

- **Incorpora video:** presenta il player per la visualizzazione del video.
- **Incorpora poster:** presenta l'immagine di copertina del video; all'immagine può essere associato un link personalizzato, ad esempio un collegamento per raggiungere il video vero e proprio (un uso tipico si ha quando in una pagina si desidera presentare un elenco di video fra cui scegliere senza per questo presentare direttamente un elenco di player).

- **Widget Live**

- **Incorpora diretta:** presenta il player per la visualizzazione della diretta.

REQUISITI E SPECIFICHE DI SISTEMA

- **Incorpora poster:** presenta l'immagine di copertina della diretta; analogamente a quanto già detto per i video, all'immagine può essere associato un link personalizzato che rimanda alla diretta vera e propria (collegamento verso altra pagina del sito ospite).
- **Widget Playlist**
 - **Incorpora playlist:** presenta il player per la visualizzazione della playlist di video.
 - **Incorpora poster:** presenta l'immagine di copertina della playlist; analogamente a quanto già detto per i video, all'immagine può essere associato un link personalizzato che rimanda alla playlist vera e propria (collegamento verso altra pagina del sito ospite).
- **Widget Canale**
 - **Incorpora canale:** presenta un video, un live o una playlist a seconda di cosa è stato pianificato per la trasmissione nel momento in cui l'utente vi accede.
 - **Incorpora poster:** presenta l'immagine di copertina associata al canale; analogamente a quanto già detto per i video, all'immagine può essere associato un link personalizzato che rimanda al canale vero e proprio (collegamento verso altra pagina del sito ospite).
- **Widget Radio**
 - **Incorpora radio:** presenta il player per l'ascolto dello streaming radio.
- **Widget Archivio:** offre la possibilità di incorporare in un sito una visualizzazione paginata di tutti i video pubblicati da un determinato utente proprietario ed eventualmente ristretti ad una o più categorie (al momento non è disponibile una interfaccia grafica per la generazione del relativo script).
- **Widget Ricerca:** offre la possibilità di effettuare una ricerca sull'archivio dei filmati pubblicati da un determinato utente proprietario ed eventualmente limitatamente ad una o più categorie selezionate (al momento non è disponibile una interfaccia grafica per la generazione del relativo script).

Schede tecniche dei widget

Tanti parametri per personalizzare il risultato finale.

Di seguito si riportano le indicazioni tecniche per l'utilizzo degli script dei widget.

Per i widget *video*, *diretta*, *playlist*, *canale* e *radio* fare riferimento alla pagina di dettaglio della risorsa stessa presente sulla piattaforma. Tutte le opzioni sono infatti configurabili e l'interfaccia consente la costruzione del widget secondo le proprie esigenze.

Widget Archivio

Incorpora l'archivio dei tuoi video sulle pagine di siti e portali.

Consente la visualizzazione paginata dei video pubblicati, eventualmente con restrizione ad una o più categorie selezionate. Lo script ha il seguente aspetto:

```
<script type="text/javascript"  
src="http://stream.senape.tv/widget/archive.action?v=2.0&uid=US  
ERID&wdt=190&hgt=130&cmp=wlp,tll&cat="></script>
```

I parametri che possono essere specificati sull'url dello script sono:

- **cat** = uno o più codici di categoria separati da virgola (i codici sono numerici e possono essere reperiti consultando i dettagli delle categorie create con la propria utenza)
- **dvo** = data nella quale il video è stato realizzato, nel formato gg/mm/aaaa hh:mm (la data di realizzazione viene dedotta dall'omonimo attributo del video; nel caso non sia valorizzata si fa riferimento alla data di pubblicazione o in ultima alternativa alla data di inserimento del video)
- **dvs** e **dve** = intervallo di date all'interno del quale il filmato è stato realizzato (formato gg/mm/aaaa hh:mm)

REQUISITI E SPECIFICHE DI SISTEMA

- **wdt** = larghezza dell'immagine di anteprima
- **hgt** = altezza dell'immagine di anteprima
- **lnk** = link da associare ad ognuna delle immagini di anteprima presentate, deve essere un url che porta ad una pagina alla quale vengono passati in GET 2 parametri: l'id del video e la chiave dell'utente. Un esempio può essere: `http://miosito/video.php?id={0}&key={1}` dove {0} e {1} vengono sostituiti dinamicamente a tempo di risoluzione dello script.
- **rpp** = numero di video restituiti per pagina
- **lpl** = link per la pagina di paginazione. La visualizzazione paginata produce a piè di pagina dei link per paginare il contenuto e questo indirizzo è quello cui inoltrare la richiesta di visualizzazione delle pagine successive
- **lvp** = numero della pagina da visualizzare
- **cmp** = informazioni (in ordine) da presentare relativamente ad ogni video; i valori ammessi sono:
 - **wlp**: immagine anteprima
 - **ttr**: titolo
 - **abs**: descrizione
 - **cat**: elenco categorie associate
 - **dur**: durata
 - **own**: proprietario
 - **rec**: data realizzazione/registrazione
 - **pub**: data pubblicazione/rilascio
- **typ** = tipologia dei contenuti da selezionare. In maniera predefinita i contenuti selezionati sono soltanto quelli pubblicati; è possibile variare questo comportamento valorizzando tale parametro al valore **arc** per restituire i soli video col flag "archivio" o **hig** affinché vengano restituiti i soli video che hanno il flag "in evidenza"
- **ord** = dato su cui ordinare i risultati, uno fra i seguenti: **ttr** (titolo), **dur** (durata), **ins** (data inserimento), **upd** (data ultimo aggiornamento), **rec** (data realizzazione), **rel** (data pubblicazione)

REQUISITI E SPECIFICHE DI SISTEMA

- **asc** = ordinamento dei risultati: **true** per ascendente, **false** per discendente

Il codice html risultante utilizza delle classi (attributo html *class*) per ogni elemento DOM generato per cui è possibile associare uno stile nel foglio css del sito ospite per poter formattare a piacere immagini, testi, posizione.

L'esecuzione di questo widget è limitata al dominio di appartenenza dell'utente proprietario, ad esempio: sitoutente.it.

Widget Ricerca

Incorpora la ricerca dei tuoi video sulle pagine di siti e portali.

Del tutto analogo al precedente widget Archivio, consente la visualizzazione paginata dei video in archivio, eventualmente con restrizione ad una o più categorie selezionate in risposta ad una interrogazione basata su un termine libero di ricerca.

Lo script ha il seguente aspetto:

```
<script type="text/javascript"
src="http://stream.senape.tv/widget/search.action?v=2.0&uid=USE
RID&wdt=190&hgt=130&cmp=wlp,t1,cat,rec,pub&cat=&trm=&ord=rec">
</script>
```

I parametri che possono essere specificati sull'url dello script sono:

- **trm** = parola o parole separate da spazio da ricercare negli attributi del video (titolo, descrizione, parola chiave, ecc)
- **tar** = tipo di elemento a cui restringere la ricerca; i valori ammessi sono: **video**, **live** o **both** (default)

Il codice html risultante utilizza delle classi (attributo html *class*) per ogni elemento DOM generato per cui è possibile associare uno stile nel foglio css del sito ospite per poter formattare a piacere immagini, testi, posizione.

L'esecuzione di questo widget è limitata al dominio di appartenenza dell'utente proprietario, ad esempio: sitoutente.it.

Le API

Con semplici chiamate HTTP puoi arricchire le tue pagine e le tue app.

La piattaforma può essere interrogata tramite l'invocazione di API appositamente implementate per consentire agli sviluppatori di avere un maggior controllo sulla presentazione dei contenuti della piattaforma all'interno di portali o di app.

In particolare le API restituiscono una rappresentazione **JSON** dei contenuti della piattaforma di un certo utente (individuato dal parametro obbligatorio **uid** specificato nella chiamata). Tale rappresentazione consente ad uno sviluppatore di utilizzare arbitrariamente le informazioni per la costruzione di elementi html personalizzati: ad esempio la presentazione di un carosello per la selezione dei video, la presentazione di un elenco di titoli di filmati e così via.

Ricerca e selezione video

Cercare video attraverso una chiamata HTTP.

L' invocazione ha il seguente aspetto:

```
http://stream.senape.tv/api/search.action?v=2.0&uid=USERID
```

Ai parametri **v** e **uid** che sono obbligatori, possono essere aggiunti, a seconda delle esigenze, i seguenti (in GET sulla chiamata):

- **cat** = valore numerico singolo o molteplice separato da virgola per restituire i soli video di una determinata categoria (i codici sono numerici)

REQUISITI E SPECIFICHE DI SISTEMA

e possono essere reperiti consultando i dettagli delle categorie create con la propria utenza)

- **dvo** = data nella quale il video è stato realizzato, nel formato gg/mm/aaaa hh:mm (la data di realizzazione viene dedotta dall'omonimo attributo del video; nel caso non sia valorizzata si fa riferimento alla data di pubblicazione o in ultima alternativa alla data di inserimento del video)
- **dvs** e **dve** = intervallo di date all'interno del quale il filmato è stato realizzato (formato gg/mm/aaaa hh:mm)
- **trm** = parola o parole separate da spazio da ricercare negli attributi del video (keyword, titolo, descrizione, ecc...)
- **rpp** = numero di elementi da restituire ("record per pagina")
- **lvp** = numero della pagina da visualizzare (da utilizzare in abbinamento al parametro precedente per una visualizzazione paginata dei contenuti)
- **tar** = tipo di elemento a cui restringere la ricerca; i valori ammessi sono: **video**, **live** o **both** (default)
- **typ** = tipologia dei contenuti da selezionare. In maniera predefinita i contenuti selezionati sono soltanto quelli pubblicati; è possibile variare questo comportamento valorizzando tale parametro al valore **arc** per restituire i soli video col flag "archivio" o **hig** affinché vengano restituiti i soli video che hanno il flag "in evidenza"
- **ord** = opzione per ordinamento dei risultati; può essere valorizzato con uno dei seguenti valori: **ttl** (titolo), **dur** (durata), **ins** (data inserimento), **upd** (data ultimo aggiornamento), **rec** (data produzione/registrazione), **rel** (data pubblicazione/rilascio)
- **asc** = determina se l'ordinamento deciso col parametro precedente debba essere discendente (**false**) o ascendente (**true**)

Valore di ritorno

La chiamata restituisce un array di dati *video*, per ogni video sono presenti tutte le informazioni salienti del filmato, incluso l'array delle categorie cui appartiene:

```
[{"archive": "S", "categories": [{"description": "Top Video", "id": 1079, "representativeVideoId": null}], "contentDescription": "", "correlated": [], "description": "Prova
```

REQUISITI E SPECIFICHE DI SISTEMA

```
Friuli", "duration":102000, "highlight": "N", "id": "GNLEE91I9I50", "landing": "N", "publish": "S", "recordingDate": null, "releaseDate": "2012-10-19T00:00:00", "title": "Mirandola", "type": "V", "validityEnd": null, "validityStart": null},
```

...]

In caso di errore viene restituita una stringa contenente un messaggio esplicativo dell'errore.

NOTA

Al rilascio di nuove versioni della piattaforma è possibile che vengano aggiunti nuovi attributi nell'oggetto javascript. È compito dello sviluppatore utilizzare i soli attributi di proprio interesse dell'oggetto JSON mantenendo così la compatibilità con gli sviluppi futuri.

Reperimento dettagli video

Recuperare i dettagli di un video attraverso una chiamata HTTP.

L'invocazione ha il seguente aspetto:

```
http://stream.senape.tv/api/video.action?v=2.0&uid=USERID&rid=IDVIDEO
```

Oltre ai parametri **v** e **uid** che sono obbligatori, occorre specificare in GET sulla chiamata l'identificativo del video o del live per il quale si esegue l'interrogazione.

Valore di ritorno

```
{"archive": "S", "categories": [{"description": "Top Video", "id": 1079, "representativeVideoId": null}], "contentDescription": "", "correlated": [], "description": "di Villafranca", "duration": 104240, "highlight": "N", "id": "NMLEER2L9IMO", "landing": "N", "publish": "S", "recordingDate": null, "releaseDate": "2012-10-19T00:00:00", "title": "La nuova chiesa", "type": "V", "validityEnd": null, "validityStart": null}
```

In caso di errore viene restituita una stringa contenente un messaggio esplicativo dell'errore.

NOTA

Al rilascio di nuove versioni della piattaforma è possibile che vengano aggiunti nuovi attributi nell'oggetto javascript. È compito dello sviluppatore utilizzare i soli attributi di proprio interesse dell'oggetto JSON mantenendo così la compatibilità con gli sviluppi futuri.

Reperimento dettagli categorie

Recuperare categorie attraverso una chiamata HTTP.

L'invocazione ha il seguente aspetto:

```
http://stream.senape.tv/api/category.action?v=2.0&uid=USERID&rid=IDCATEGORIA
```

Oltre ai parametri **v** e **uid** che sono obbligatori, è possibile specificare in GET sulla chiamata, col parametro **rid**, l'identificativo di una categoria specifica per la quale restituire i dettagli. Se il parametro *rid* non viene valorizzato o è mancante viene restituita la lista di tutte le categorie disponibili per l'utente individuato dalla chiave utilizzata nell'interrogazione.

Valore di ritorno

La chiamata restituisce un oggetto *categoria*:

```
{"description": "Top Video", "id": 1079, "representativeVideoId": null}
```

In caso di errore viene restituita una stringa contenente un messaggio esplicativo dell'errore.

NOTA

Al rilascio di nuove versioni della piattaforma è possibile che vengano aggiunti nuovi attributi nell'oggetto javascript. È compito dello sviluppatore utilizzare i soli attributi di proprio interesse dell'oggetto JSON mantenendo così la compatibilità con gli sviluppi futuri.

Reperimento dettagli programmazione canale

Recuperare i dettagli di una programmazione lineare su un canale attraverso una chiamata HTTP.

L'invocazione ha il seguente aspetto:

```
http://stream.senape.tv/api/channel.action?v=2.0&uid=USERID&rid=IDCANALE
```

Il parametro **rid** individua il canale per il quale si vogliono ottenere i dati sulla programmazione.

Se non si specificano altri parametri viene restituita l'informazione sulla risorsa attualmente in programma sul canale (il valore `onAirDate` nel JSON non è presente in questo caso).

È anche possibile specificare ulteriori 2 parametri per avere come risultato l'elenco delle risorse programmate in un certo intervallo di date. I parametri in questione sono:

- **dvs** = data di inizio del periodo (nel formato gg/mm/aaaa hh:mm)
- **dve** = data di fine del periodo (nel formato gg/mm/aaaa hh:mm)

Se specificati, i parametri temporali devono essere entrambi presenti.

Valore di ritorno

La chiamata restituisce un array in ordine temporale delle risorse pianificate per la trasmissione sullo specifico canale individuato:

```
[{"onAirDate": "2012-11-01T00:00:00", "publish": "S", "description": "State seguendo la diretta della programmazione satellitare", "resourceId": "0QBXFQ2L0I5P", "title": "Diretta Canale SAT", "resourceType": "V", "resourceValidityEnd": null, "resourceValidityStart": null, "end": "2012-11-30T23:55:00", "start": "2012-09-
```

REQUISITI E SPECIFICHE DI SISTEMA

```
02T17:00:00", "videoType": "L"}, {"onAirDate": "2012-11-04T13:30:00", "publish": "S", "description": "L'incontro di Prima Divisione girone A Treviso - San Marino valido per la decima giornata di campionato. Domenica 04/11/2012 in diretta dalle ore 15.00", "resourceId": "IZOZE9GIQI58", "title": "Treviso - San Marino", "resourceType": "V", "resourceValidityEnd": null, "resourceValidityStart": null, "end": "2012-11-04T17:10:00", "start": "2012-11-04T13:30:00", "videoType": "L"},
```

...]

In caso di errore viene restituita una stringa contenente un messaggio esplicativo dell'errore.

NOTA

Al rilascio di nuove versioni della piattaforma è possibile che vengano aggiunti nuovi attributi nell'oggetto javascript. È compito dello sviluppatore utilizzare i soli attributi di proprio interesse dell'oggetto JSON mantenendo così la compatibilità con gli sviluppi futuri.

Accesso diretto all'immagine associata al singolo filmato/live

Recuperare l'immagine di copertina di un video attraverso una chiamata HTTP.

Dato l'identificativo di video o live l'immagine associata può essere reperita come risorsa statica all'url:

<http://stream.senape.tv/statics/video/wall/IDVIDEO.jpg>

In alternativa una immagine di dimensioni ridotte è reperibile analogamente all'url:

<http://stream.senape.tv/statics/video/thumb/IDVIDEO.jpg>

Web Service

Un servizio per integrazioni lato server con Senape.tv.

La piattaforma espone in modalità sicura (https) la possibilità di effettuare delle chiamate a metodi specifici pensati per consentire ad uno sviluppatore di personalizzare l'integrazione con la piattaforma.

L'URL per l'accesso al WS è il seguente:

```
https://stream.senape.tv/services/webtvPort
```

Il relativo WSDL con la documentazione dei metodi disponibili è ottenibile all'URL:

```
https://stream.senape.tv/services/webtvPort?wsdl
```

I metodi attualmente disponibili sono:

- **getSecureToken:** ritorna un token valido per la visualizzazione di una certa risorsa
- **getVideoUrl:** ritorna un URL diretto per accesso al video/live specifico per un certo tipo di dispositivo: rtmp (Flash), cupertino (iOS), Android.

Eventualmente può essere richiesta la restituzione contestuale di un token valido per la visualizzazione della risorsa stessa (da utilizzare assieme all'url restituito)

- **importVideo:** consente l'importazione su piattaforma di un video previamente caricato in una area ftp oppure inviato direttamente tramite la chiamata stessa del metodo (l'area ftp viene configurata solo dietro richiesta dell'utente)
- **removeVideo:** effettua la rimozione di un video dalla piattaforma

REQUISITI E SPECIFICHE DI SISTEMA

- **getCategories:** ritorna l'elenco delle categorie disponibili per l'utente – eventualmente ritorna i dati di una sola categoria se si specifica il relativo identificativo
- **getVideos:** ritorna i dati dei video disponibili per l'utente ricercati secondo vari criteri disponibili – eventualmente ritorna i dati di un singolo video se si specifica il relativo identificativo
- **getChannelPlanning:** ritorna la sequenza delle risorse programmate sul canale specificato a partire dall'istante della richiesta o in un intervallo di date specificato

Il Pushing

Mantenere una replica locale allineata con i contenuti presenti su Senape.tv.

Quando l'integrazione con la piattaforma richieda un elevato grado di libertà da parte dell'utilizzatore è possibile fare in modo che la piattaforma invii al sito dell'utente una notifica automatica di tutte le operazioni che avvengono sui contenuti della piattaforma (inserimenti, modifiche e cancellazioni). In tal modo un utente è in grado di mantenere una propria base dati locale che ricalca le informazioni presenti sulla piattaforma (video, dirette, categorie, programmazione dei canali) sulla quale sviluppare un sito in completa libertà.

Funzionamento

Attivare il meccanismo di PUSH.

L'attivazione del servizio di **pushing** richiede la definizione da parte del cliente di un URL al quale verranno inviate le varie notifiche tramite POST HTTP e la definizione di una password condivisa fra le due parti per la validazione delle chiamate, ovvero per poter stabilire che la chiamata HTTP proviene effettivamente dalla piattaforma Senape.

La chiamata prevede l'invio dei seguenti parametri in ogni chiamata POST:

- **type**: indicazione sulla tipologia del contenuto inviato nel campo **data**
- **data**: contenuto in formato JSON delle modifiche avvenute lato piattaforma
- **timestamp**: timestamp di generazione (invio) della richiesta nel formato: gg/mm/aaaa hh:mm:ss.MMM

REQUISITI E SPECIFICHE DI SISTEMA

- **version:** numero di versione del formato dei dati (es. "2.0")
- **hash:** valore per la validazione della chiamata. Tale valore corrisponde al calcolo del valore MD5 effettuato sulla stringa risultante dalla seguente concatenazione di valori:

type+lunghezza_campo_data + timestamp + versione +
password_convenuta

Alla ricezione di una notifica PUSH il destinatario può validare la chiamata effettuando lo stesso computo sui dati del POST assieme alla password in proprio possesso e verificando che l'MD5 calcolato sulla stringa composta corrisponda effettivamente al valore dell'hash ricevuto con la notifica.

Il POST ha sempre encoding UTF-8.

La risposta attesa dalla piattaforma a conferma dell'avvenuta gestione della chiamata HTTP deve rispondere alle seguenti caratteristiche:

- avere codice HTTP di ritorno uguale a 200 (OK)
- avere contenuto della risposta (body) in formato testo riportante esplicitamente la parola "OK" in caso di successo o "KO" per segnalare la necessità di ricevere nuovamente la notifica

Ogni altro codice HTTP (es. 404, 503, ...) ed ogni contenuto diverso da "OK" all'interno di una risposta 200 vengono assimilati ad un fallimento della comunicazione che viene dunque ripetuta entro il minuto successivo, fino al suo buon esito.

Strutture dei dati

Gestire i risultati di una chiamata PUSH.

Il campo **data** può avere un contenuto variabile di volta in volta in relazione alle modifiche che vengono segnalate. In particolare esistono 3 tipologie di elementi che possono essere contenuti in una notifica e sono riconoscibili tramite i seguenti 3 valori previsti per il parametro type del POST:

- **VID:** il contenuto JSON corrisponde ad un oggetto video. Il video può rappresentare sia un contenuto on-demand che un contenuto "live", la distinzione fra queste due eventualità è determinata dal valore

REQUISITI E SPECIFICHE DI SISTEMA

dell'attributo `type` dell'oggetto JSON il quale assume valore "L" in caso di video "live" e "V" in caso di contenuto on-demand.

- **CAT:** il contenuto JSON corrisponde ad un oggetto categoria.
- **CHP:** il contenuto JSON corrisponde ad un oggetto programmazione canale.

Tutti gli oggetti JSON possiedono l'attributo `changeType` che indica la tipologia del cambiamento avvenuto sull'oggetto e sta ad indicare a seguito di quale evento la piattaforma ha inviato la notifica PUSH. Sono previsti i seguenti valori:

- **I** (Insert): l'oggetto è nuovo ed è stato creato sulla piattaforma
- **U** (Update): l'oggetto è stato modificato sulla piattaforma
- **D** (Delete): l'oggetto è stato eliminato dalla piattaforma
- **R** (Reset): previsto per la risincronizzazione delle informazioni, viene utilizzato per richiedere di ripristinare l'oggetto esattamente allo stato comunicato. È compito dello sviluppatore considerare se questa operazione richieda di fatto un inserimento o un aggiornamento dei dati.

Gli attributi previsti per i vari oggetti JSON sono i seguenti.

Video

- **id:** identificativo (stringa di 12 caratteri)
- **title:** titolo (stringa di 255 caratteri)
- **description:** descrizione (stringa di 4000 caratteri)
- **type:** tipologia del video ("V" in caso di video on-demand, "L" in caso di diretta live)
- **duration:** durata (valore numerico espresso in millisecondi)

Si segnala che per una diretta (live) la durata è sempre 0 (zero) poiché non esiste un vincolo effettivo alla durata di una trasmissione in diretta. Una volta terminata, una diretta può essere convertita in video e quindi resa disponibile per la fruizione on-demand: in tal caso la durata viene dedotta dal video registrato sul cloud dalla piattaforma e comunicata tramite una notifica successiva PUSH.

REQUISITI E SPECIFICHE DI SISTEMA

Anche al caricamento di un nuovo video la durata è provvisoriamente 0 (zero) fino a che questo non viene preso in carico dalla piattaforma per la conversione; non appena rilevata la durata del video sorgente caricato, viene inviata una notifica PUSH contenente la nuova informazione.

- **validityStart**: data e ora di inizio validità del video
- **validityEnd**: data e ore di fine validità del video
- **categories**: array di categorie cui è associato il filmato; per ogni categoria è riportato l'identificativo **id** (numerico) e il nome **description** (stringa di 255 caratteri)
- **correlated**: array di identificativi di filmati correlati
- **publish**: flag pubblicato (S o N)
- **archive**: flag archivio (S o N)
- **highlight**: flag "in evidenza" (S o N)
- **landing**: flag visibile su landing page dell'utente (S o N)
- **fullAvailability**: flag effettiva disponibilità del filmato convertito (S o N)
- **recordingDate**: data realizzazione
- **releaseDate**: data pubblicazione
- **notes**: note interne (stringa 4000 caratteri)
- **changeType**: tipologia del cambiamento avvenuto sull'oggetto per il quale si riceve la notifica (1 carattere).

Esempio concreto:

```
{ "archive": "S", "categories": [ { "description": "Sport", "id": 1142 } ],  
  "changeType": "I", "correlated": [ "ANPFEQ1I9L67" ], "description":  
  San Marino calcio l'analisi del  
  Lunedì", "duration": 103253, "fullAvailability": "S", "highlight": "N",  
  "id": "5MOEEQ2LRIM8", "landing": "N", "notes": null, "publish": "S",  
  "recordingDate": "2012-10-  
  15T13:00:00", "releaseDate": null, "title": "San Marino calcio  
  l'analisi del  
  Lunedì", "type": "V", "validityEnd": null, "validityStart": null }
```

Categoria

- **id**: identificativo (numerico long)
- **description**: nome categoria (stringa di 255 caratteri)
- **representativeVideoId**: identificativo di un video rappresentativo della categoria
- **changeType**: tipologia del cambiamento avvenuto sull'oggetto per il quale si riceve la notifica (1 carattere).

Esempio concreto:

```
{"changeType": "U", "description": "Area  
News", "id": 1204, "representativeVideoId": null}
```

Programmazione canale

La programmazione di un canale è sempre costituita da una sequenza di contenuti che vengono programmati per apparire su un canale in una certa fascia oraria. Tali contenuti individuano per un certo canale la successione di una serie di orari ai quali cambia la risorsa in programmazione.

In maniera predefinita la piattaforma invia la sequenza dei programmi pianificati dalla data odierna (quella in cui avviene la notifica) fino ai 90 giorni successivi.

- **channelId**: identificativo del canale della piattaforma (stringa di 12 caratteri)
- **changeType**: tipologia del cambiamento avvenuto sull'oggetto per il quale si riceve la notifica. Si tratta sempre del valore R (Reset) in quanto ogni aggiornamento va a sostituire l'elenco dei programmi precedenti.
- **plannings**: array degli elementi in programmazione sul canale.

Ognuno degli elementi in programmazione viene individuato dalle seguenti informazioni:

- **onAirDate**: data e ora della messa in onda sul canale (non presente in caso di restituzione della sola trasmissione corrente)

REQUISITI E SPECIFICHE DI SISTEMA

- **resourceId**: identificativo della risorsa programmata (stringa 12 caratteri).

L'identificativo in questione può fare riferimento ad una qualsiasi delle risorse disponibili sulla piattaforma che può essere trasmessa su un canale. Può dunque trattarsi semplicemente di un video (on-demand o live) ma anche di una playlist.

- **resourceTitle**: titolo della risorsa programmata (stringa di 255 caratteri)
- **resourceDescription**: descrizione della risorsa programmata (stringa di 4000 caratteri)
- **resourceType**: tipologia della risorsa programmata (1 carattere: "V" in caso di video o "P" in caso di playlist)
- **videoType**: tipologia video; tale attributo è valorizzato solo nel caso la tipologia della risorsa sia un video e consente di distinguere la tipologia del video programmato fra on-demand e live (1 carattere, "V" in caso di video on-demand o "L" in caso di live)

La presenza di eventuali elementi "nulli" sta ad indicare la fine temporale dell'elemento precedente (l'elemento nullo presenta i soli valori di **onAirDate** - impostato alla data e ora di fine dell'elemento precedente - e **resourceType** impostato a "U")

Esempio concreto:

```
{ "changeType": "R", "channelId": "RHCRDQ1IQIM8", "plannings": [ { "onAirDate": "2012-11-28T00:00:00", "resourceDescription": "State seguendo la diretta della programmazione satellitare", "resourceId": "0QBXFQ2L0I5P", "resourceTitle": "Diretta Canale SAT", "resourceType": "V", "videoType": "L" }, { "onAirDate": "2013-02-25T06:55:00", "resourceDescription": null, "resourceId": null, "resourceTitle": null, "resourceType": "U", "videoType": null } ] }
```

NOTA

Al rilascio di nuove versioni della piattaforma è possibile che vengano aggiunti nuovi attributi agli oggetti javascript. È compito dello sviluppatore utilizzare i soli attributi di proprio interesse dell'oggetto JSON mantenendo così la compatibilità con gli sviluppi futuri.

Inizializzazione della base dati e successive sincronizzazioni

Buone prassi per un corretto allineamento.

L'utente che voglia implementare la ricezione di notifiche push avrà bisogno di inizializzare la propria base dati allineandola alle informazioni eventualmente già presenti sulla piattaforma.

In questo caso è possibile scaricare dalla piattaforma in un unico file una situazione completa, sempre in formato JSON, di tutti i contenuti della piattaforma. Tale operazione può essere ripetuta in un qualsiasi momento, ad esempio allo scopo di riallineare eventuali discrepanze che dovessero essersi generate accidentalmente nel tempo.

Tale file è strutturato nel seguente modo:

- **categories:** array delle categorie esistenti (vedere il singolo oggetto JSON **Categoria** per quanto riguarda gli attributi; in particolare il campo **changeType** è valorizzato a R (reset) ad indicare la necessità di sostituire tutte le informazioni eventualmente preesistenti con quelle inviate
- **videos:** array dei video e live esistenti (vedere il singolo oggetto JSON **Video** per quanto riguarda gli attributi; in particolare il campo **changeType** è valorizzato a R (reset) ad indicare la necessità di sostituire tutte le informazioni eventualmente preesistenti con quelle inviate
- **programs:** array delle programmazioni esistenti su tutti i canali dell'utente (vedere il singolo oggetto JSON **Programmazione canale** per quanto riguarda gli attributi; in particolare il campo **changeType** è valorizzato a R (reset) ad indicare la necessità di sostituire tutte le informazioni eventualmente preesistenti con quelle inviate.

Esempio concreto:

```
{ "categories": [ { "changeType": "R", "description": "Cinema", "id": 1002, "representativeVideoId": null }, { "description": "In vetrina", "id": 1198 }, { "description": "Programmi", "id": 1240 } ], "changeType": "R", "correlated": [], "description": "A quasi 40 anni dalla dittatura in
```

REQUISITI E SPECIFICHE DI SISTEMA

```
Argentina", "duration":1935000, "fullAvailability":"S", "highlight": "N", "id": "T7IZE0GIRL67", "landing": "N", "notes": null, "publish": "S", "recordingDate": "2012-10-19T22:00:00", "releaseDate": null, "title": "Restituiti", "type": "V", "validityEnd": null, "validityStart": null}, ...]
```

L'importazione "trasparente" dei video

Come implementare una procedura di import dei propri video.

Nel caso un utente voglia implementare autonomamente il caricamento dei contenuti video tramite il proprio sito, nascondendo completamente all'utente ogni riferimento alla piattaforma in uso, è possibile abilitare e seguire la seguente procedura:

1. Il cliente decide autonomamente la modalità con la quale i suoi utenti caricano i contributi video. Non c'è quindi la necessità di far accedere in qualche modo un utente alla piattaforma Senape né di fargli utilizzare il programma Senape Upload.
2. Ogni file caricato dagli utenti dovrà essere trasferito via FTP verso un server della piattaforma Senape utilizzando le credenziali convenute e fornite allo scopo.
3. Una volta che il file è stato caricato sull'FTP occorre invocare una action specifica della piattaforma Senape (URL: `http://stream.senape.tv/exec/import.action`) passando alcuni parametri in POST:
 - **do**: valore costante "importVideo"
 - **user**: valore costante corrispondente allo username del cliente sulla piattaforma
 - **file**: indica il nome del file che è stato caricato sull'FTP e che si vuole importare.

Il nome del file deve rispettare un formato prestabilito così definito:

REQUISITI E SPECIFICHE DI SISTEMA

nome-file-originale-inclusivo-di-estensione + “_” + suffisso

Il suffisso deve essere appeso al nome del file e separato da questo dal carattere underscore, compito del suffisso è quello di prevenire una possibile collisione dovuta al caricamento di due file con lo stesso nome da parte di due utenti diversi.

Si suggerisce pertanto di utilizzare come suffisso un numero random o un timestamp in millisecondi.

Esempio di un nome di file corretto: “prova.mp4_123123123”

- **title**: titolo da assegnare al video una volta importato sulla piattaforma
 - **descr** (facoltativo): descrizione da assegnare al video una volta importato sulla piattaforma
 - **categ** (facoltativo): lista separata da virgola dei codici numerici delle categorie da associare al filmato (i codici da utilizzare devono essere noti a priori e reperiti fra quelli disponibili per l'utente sulla piattaforma)
 - **recdt** (facoltativo): data della registrazione nel formato gg/mm/aaaa hh:mm:ss
 - **reltdt** (facoltativo): data della pubblicazione nel formato gg/mm/aaaa hh:mm:ss
 - **pub** (facoltativo): impostazione del flag pubblicato (default = false)
 - **lan** (facoltativo): impostazione del flag visibilità nella pagina personale (default = false)
 - **arc** (facoltativo): impostazione del flag visibilità in archivio (default = false)
 - **hig** (facoltativo): impostazione del flag visibilità in evidenza (default = false)
4. La piattaforma risponde con l'invio di un risultato JSON avente i seguenti attributi:
- **code**: “OK” o “KO” a seconda che l'importazione sia andata o meno a buon fine

REQUISITI E SPECIFICHE DI SISTEMA

- **message:** in caso di successo contiene l'ID che la piattaforma ha assegnato al video, tale informazione deve essere memorizzata e utilizzata per comporre dinamicamente il codice dei widget (es. il widget Video per mostrare il video caricato)
5. La piattaforma si fa carico della conversione del video caricato; l'utenza Senape.tv riceverà notifica via e-mail delle avvenute conversioni e il filmato potrà essere gestito come ogni altro fra quelli disponibili.

La stessa funzionalità di importazione può essere utilizzata anche per aggiornare un video oltre che per inserirlo. In questo caso occorre specificare fra i parametri della chiamata anche il parametro id valorizzato con l'identificativo del video che si vuole andare a sostituire sulla piattaforma.

Le statistiche

Panoramica sulle statistiche di visualizzazione.

Il rilevamento generale di statistiche è demandato ad implementazione da parte dell'utilizzatore di un sistema arbitrario di tracciatura delle visualizzazioni delle pagine del sito che ospitano i contenuti video (ad esempio ricorrendo a strumenti quali Google Analytics, ShinyStat, ecc.).

I dati relativi al picco massimo di visualizzazioni contemporanee realizzate da un video o da una diretta sono sempre consultabili all'interno dei dettagli della risorsa in questione.

A richiesta, su specifici eventi, possono essere forniti dal personale Senape.tv i dati relativi alle visualizzazioni effettive rilevate durante una diretta o dal momento della pubblicazione del video on-demand.

Il rilevamento autonomo di dati più accurati relativi alle effettive visualizzazioni (pressione del bottone Play) da parte degli utenti visitatori è attualmente allo studio.

Modalità accesso sicuro

Consentire l'accesso ai propri contenuti solo a sorgenti sicure.

È possibile configurare un utente in modo che tutti i suoi contenuti risultino accessibili solo in modalità sicura. Le implicazioni di questa impostazione sono le seguenti:

- L'utente deve aggiungere ai normali parametri di configurazione dei widget il parametro `sec` affinché questi funzionino.

Per recuperare un token valido occorre che l'utente effettui previamente una chiamata tramite WS all'apposito metodo che restituisce il token per la valorizzazione del parametro.

Esempio:

```
<script type="text/javascript"
src="http://stream.senape.tv/widget/channel.action?v=2.0&uid
=utenteprova&rid=9KFSFBKLAL6R&wdt=816&hgt=459&cmp=ply&ast=fa
lse&sub=false&vol=100&mut=false&pll=false&sec=SzQlEAzrRPv8GB
UCZKMEUQGv_AH7Rw7gic8nhc1Ow_sF_oNMHKx1-
hS11fX1siw4pvGwA9g0ITNMmf1sUHhwKvUjJD9sqPbIO7vSs-
vPo9I"></script>
```

- Il token generato consente l'utilizzo del widget un numero limitato di volte e per un tempo limitato.

La validità del token viene convalidata sia dalla piattaforma (web) che dal server di streaming.

Il token è richiesto per i widget video, live, playlist, channel e radio.

La piattaforma si occupa di fare in modo che i vincoli del token vengano rispettati a livello di widget.

REQUISITI E SPECIFICHE DI SISTEMA

Il server di streaming si occupa di fare in modo che i vincoli del token vengano rispettati a livello di visualizzazione di una risorsa (evento play).

Per la normale operatività e visibilità delle anteprime dei widget all'interno del back end della piattaforma un token ha un numero più elevato di utilizzi e una validità massima di 4 ore.

Lato streaming un token è ritenuto valido fino a 12 ore dalla creazione per permettere ad una pagina web di essere "funzionante" per tutto il tempo di una diretta.

Il numero di utilizzi ammesso da un token è definito da due property applicative che hanno i seguenti valori:

- 5 utilizzi in embedding
- 50 utilizzi all'interno del back end

NOTA

Al momento l'utilizzo della playlist nel back end prevede per ogni video 5 utilizzi (poiché l'anteprima sfrutta lo stesso accesso dei widget)

Il numero di utilizzi del token è indispensabile per consentire ad un player di effettuare stop e play più di una volta.

Il tempo massimo di validità del token deve essere ragionevolmente alto per fare in modo che uno stop e play possa funzionare fino alla fine di una diretta o di una playlist.

A

Adobe Flash Media Live Encoder

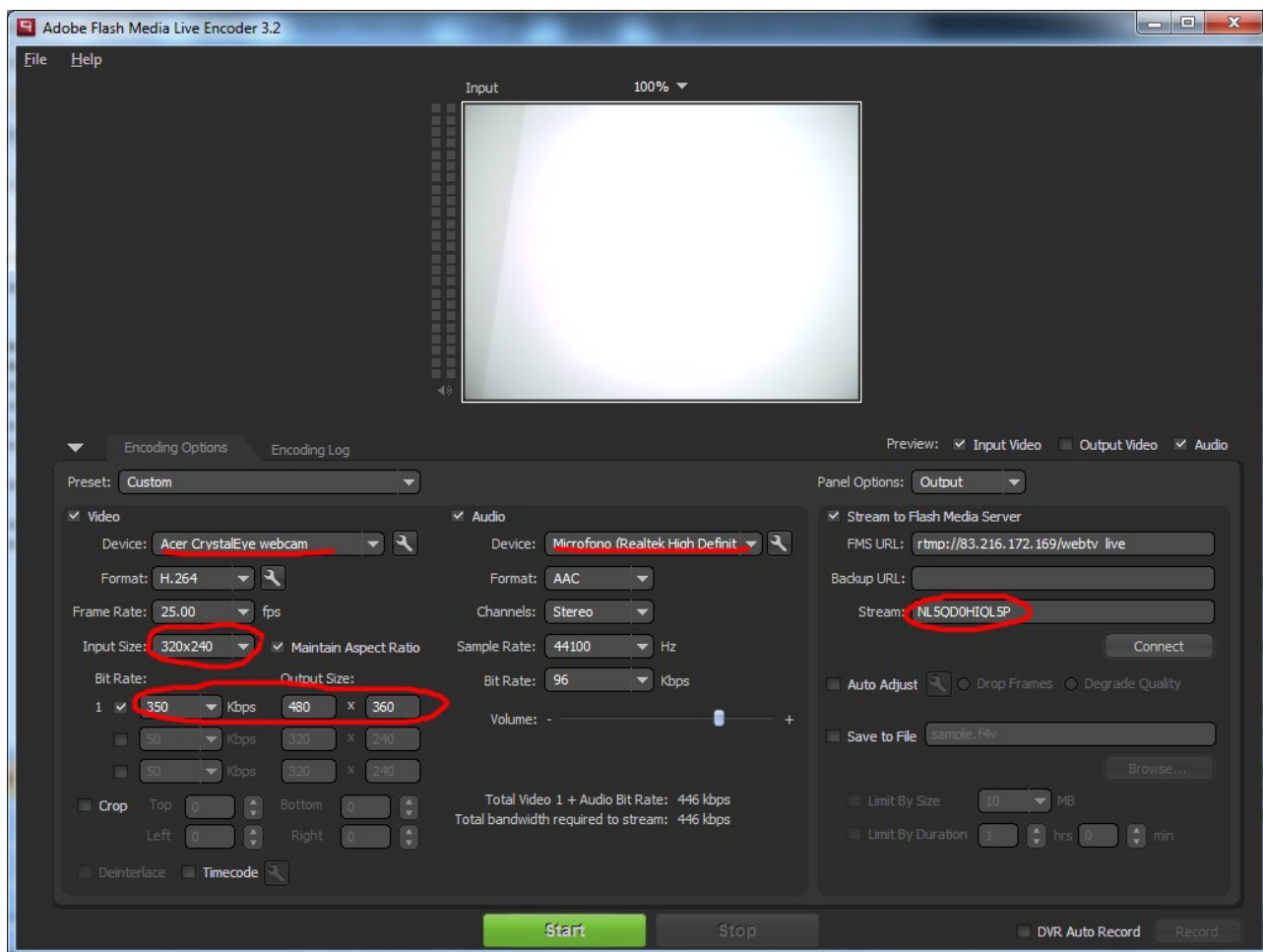
Come trasmettere le proprie dirette video.

Per la trasmissione di una diretta occorre disporre di uno strumento di cattura video (es. una telecamera, una webcam, ecc.) e di un programma specifico di codifica del flusso dati chiamato **encoder**.

L'encoder – necessario per la pubblicazione di una diretta su Senape.tv – richiede il supporto del *protocollo RTMP* pertanto si consiglia l'utilizzo di [Adobe Flash Media Live Encoder](#), software gratuito che può essere scaricato direttamente dal sito di Adobe (attualmente disponibile nella versione 3.2 per Windows e per Mac). In alternativa esistono soluzioni commerciali a pagamento – che si arricchiscono di funzionalità più evolute come, ad esempio, una regia software o la possibilità di aggiungere titoli in sovrapposizione al flusso video – quali [Wirecast](#) (attualmente disponibile nella versione 4.2 per Windows e per Mac).

Una volta installato presenta un'interfaccia abbastanza intuitiva in cui occorre configurare la periferica video, audio e l'indirizzo del server di streaming (FMS URL) che può essere rilevato consultando la configurazione del proprio profilo utente sulla piattaforma Senape.tv (es. `rtmp://o1.senape.tv/webtv_live`).

REQUISITI E SPECIFICHE DI SISTEMA



Per agevolare la configurazione di base viene fornito assieme alle credenziali del proprio utente un file xml che può essere aperto dall'applicazione encoder e che configura buona parte dei valori.

Facendo riferimento all'immagine soprastante sono evidenziati in rosso i dati che possono (o devono) essere modificati dall'utente:

- **Device video e audio** vengono proposte automaticamente fra quelle disponibili nel pc in cui è installato l'encoder, occorre selezionare ed eventualmente configurare (tramite i bottoni a lato con l'icona della chiave inglese) i dispositivi video e audio che si vogliono utilizzare come ingresso e quindi trasmettere in diretta
- L'**input Size** del video è la dimensione dell'immagine in ingresso e dipende sempre dalla telecamera in uso; la cosa importante è utilizzare delle dimensioni in ingresso che siano maggiori o al limite uguali a quelle che si impostano in uscita (ovvero nei campi immediatamente sottostanti)

REQUISITI E SPECIFICHE DI SISTEMA

- **Bit Rate** e **Output Size** del video sono da impostare ai valori predefiniti previsti dalla piattaforma, attualmente le indicazioni sono di 480 pixel di larghezza e di 270 o 360 di altezza a seconda che il video in ingresso sia rispettivamente in formato 16/9 o 4/3. Nel caso le dimensioni rilevate dalla periferica in ingresso non corrispondano al formato di 480x270 (16/9) o 480x360 (4/3) occorre rimuovere la spunta sulla voce **Maintain Aspect Ratio** e quindi immettere manualmente l'altezza desiderata.

Il bit rate è molto importante e va regolato in base alla banda in upload disponibile al momento della diretta. Solitamente si utilizza un valore compreso fra 150 e 416 facendo comunque in modo che la somma del bitrate video e di quello audio non superi complessivamente i 512 Kbit/sec (il valore totale di banda richiesto è presentato ai piedi della sezione centrale dell'applicazione, sotto le impostazioni dell'audio).

Superare tale valore può pregiudicare la possibilità per gli utenti fruitori di riuscire a seguire la diretta e compromettere la stessa pubblicazione.

- **FMS URL** è il campo dove indicare l'indirizzo del server per la pubblicazione.
- **Stream** è il campo dove riportare l'identificativo dell'evento live creato sulla piattaforma.

All'occorrenza anche il **Bit Rate** dell'audio può essere modificato ma è opportuno mantenere un valore compreso fra 96 e 128 (eccedere nella qualità dell'audio va a scapito della qualità del video poiché resta il vincolo di non superare i 512 Kbit/sec totali).

Come **Sample Rate** mantenere sempre 44100 (necessario per compatibilità con iPad, iPhone, ...).

Una nota particolare va fatta sul **Codec Audio**: il software di Adobe installato su Windows mette a disposizione solo il formato MP3 mentre per avere una buona fruibilità sui dispositivi mobili (iPhone e Android) occorre utilizzare il codec AAC (Su MAC normalmente risulta già presente).

Tale codec è acquistabile a parte da una società che rilascia un plugin specifico per l'encoder e che può essere trovato a questo indirizzo: <http://www.mainconcept.com/products/apps-plug-ins/plug-ins-for-adobe/aac-encoder.html>.

Al momento dell'avvio della trasmissione con l'encoder vengono richieste delle credenziali che sono sempre fornite assieme alle credenziali del proprio

utente della piattaforma e che solitamente coincidono con quelle per l'accesso web, eventualmente è possibile richiedere che siano differenti nel caso si vogliano condividere con altri solo quelle della piattaforma ma non quelle per la pubblicazione.

Trasmissione di una diretta radio

Come trasmettere le proprie dirette audio.

Per la trasmissione di una radio vale quanto già esposto per il video. Il nome dello stream su cui pubblicare è quello rilevabile dal proprio elemento **Radio** creato sulla piattaforma.

L'unica variazione consiste nella necessità di disabilitare la codifica video tramite l'apposito checkbox posto in alto a sinistra della sezione **Video** (la prima delle 3 colonne in cui è suddivisa l'interfaccia dell'encoder).

La qualità consigliata per l'audio è di 96 Kbit/s con codec AAC, salendo eventualmente a 128 Kbit/s in caso di MP3. Sample Rate sempre 44100 (necessario per compatibilità con iPad, iPhone, ...).

