Fitness use case

Auto-record calls when the instructor joins the class

Using meeting tokens and Daily's recording API to record a specific call participant



Jess Mitchell

Published: 31 Mar 2022 (Updated: 31 Mar 2022) • 7 min read



> *This is part three in our fitness use case series on how to build a custom Daily fitness app.*

Lately at Daily, we've been discussing lots of video call features specific to the fitness class use case. We've been covering this discussion in our fitness demo series, which is using our fitness demo app as a base for some new features.

So far in this series, we've looked at all the features the fitness demo

currently has to offer. We've also reviewed how the mute all audio and video buttons work, and how to upgrade them to mute automatically when the class begins.

In today's tutorial, we'll look at adding another automated feature when the instructor joins the class: recording!

Specifically, we'll look at:

- How to set meeting token properties so that recording automatically starts when the meeting owner joins.
- How to update our recording settings for the call so only the instructor's video and audio are recorded. This can be your gift to any fitness class attendee who does ***not*** need recorded proof of how sweaty they were in the class!

Now that we have a plan, let's get started. 💪

Running the fitness demo app locally

To achieve our feature goals for the day, we'll need to update our fitness demo app's existing features. If you prefer to follow along locally, you will need to first clone the app and install its dependencies from the root directory:

```
git clone git@github.com:daily-demos/examples.git
cd examples
yarn
```

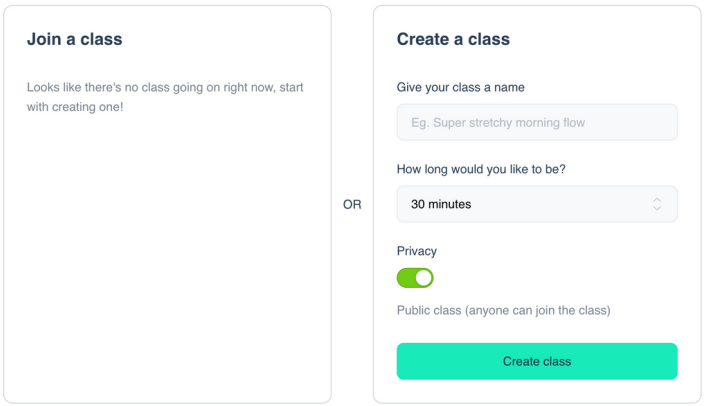
Once installed, go to the fitness demo directory and rename the example `.env` file to be used locally:

```
cd custom/fitness
mv env.example .env.local
```

You'll then need to create a Daily account, and add your Daily domain and API key to the `.env.local` file.

Now you're ready to start your local server from the root directory again:

```
cd ../..
yarn workspace @custom/fitness-demo dev
```



Fitness demo home page

Note: Check out the project's README if you're looking for more information on running the app locally.

Once here, you can join a class as a participant on the left if there are any live classes to join, or you can create a new class as an instructor on the right. We'll be pretending we're an instructor and create a new class on the right for today's tutorial.

Now that we're set up, let's move on to reviewing how Daily meeting tokens relate to recordings.

How to set up automatic recordings with Daily meeting tokens

With

With `daily-js`, you can start or stop recordings in a few different ways. You can do it programmatically after joining a call with the startRecording and stopRecording methods. In Daily Prebuilt's UI, you simply press the `Start recording` button in the bottom tray. Or, as a final option, you can start it with a meeting token. (That's what we'll be doing today!)

In our `examples` repo, a meeting token is automatically created for anyone who creates a new fitness class to distinguish them as the meeting owner. Meeting tokens are used for authenticating specific participants and can give some extra privileges too, depending on the token's settings.

```
export default async function handler(req, res) {
 const { roomName, isOwner } = req.body;

 if (req.method === 'POST' && roomName) {
  const properties = {
     room_name: roomName,
     is_owner: isOwner,
   }

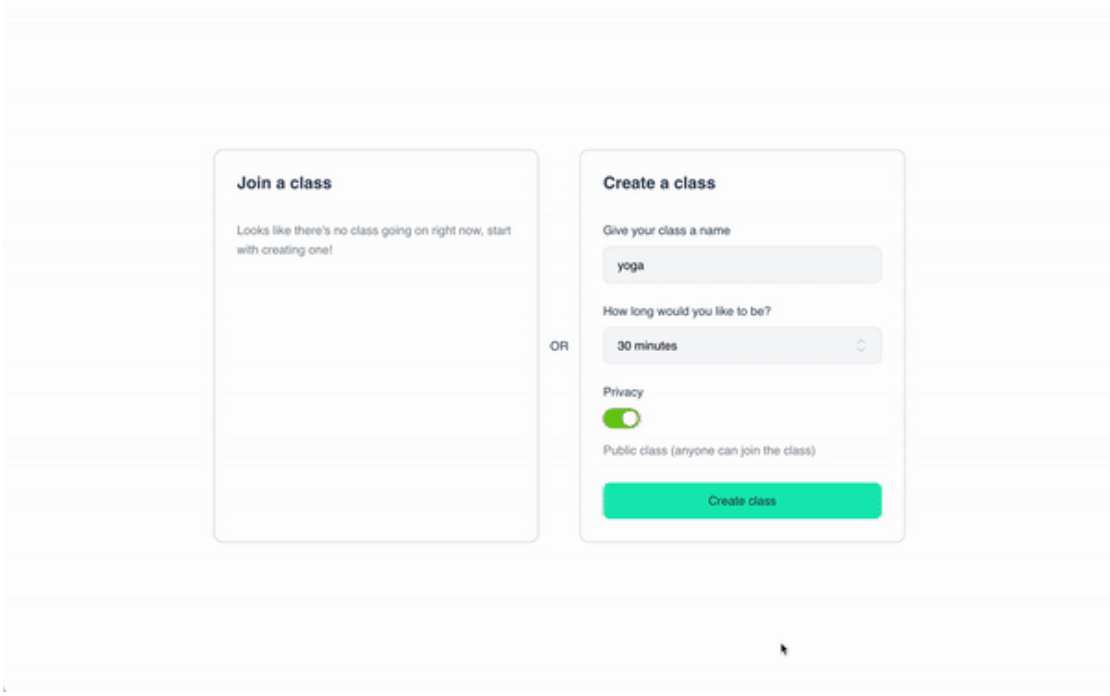
   const options = {
     method: 'POST',
     headers: {
       'Content-Type': 'application/json',
       Authorization: `Bearer ${process.env.DAILY_API_KEY}`,
     },
     body: JSON.stringify({ properties }),
   };

   const dailyRes = await fetch(
     `${
       process.env.DAILY_REST_DOMAIN || 'https://api.daily.co/v1
     }/meeting-tokens`,
     options
   );
 …
```

token.js

In the code snippet above, the `is_owner` meeting token property must be set to `true` for the participant to be considered a meeting owner.

Currently, with the existing feature set in the examples repo, the instructor has to remember to click the `Record` button to start recording the class when they join.



The fitness demo's current recording user flow

We want to improve this experience a bit because surely an instructor will forget to click the button at some point. To do this, we need to add two more properties to our owner meeting token request.

- `enable_recording` : must be set to `cloud` (a.k.a. server-side recording) for this specific feature
- `start_cloud_recording` : must be set to `true` to enable starting the recording when the meeting owner joins the class

We can update the code snippet from above to include both of these:

```
…
  const properties = {
    room_name: roomName,
    is_owner: isOwner,
  }

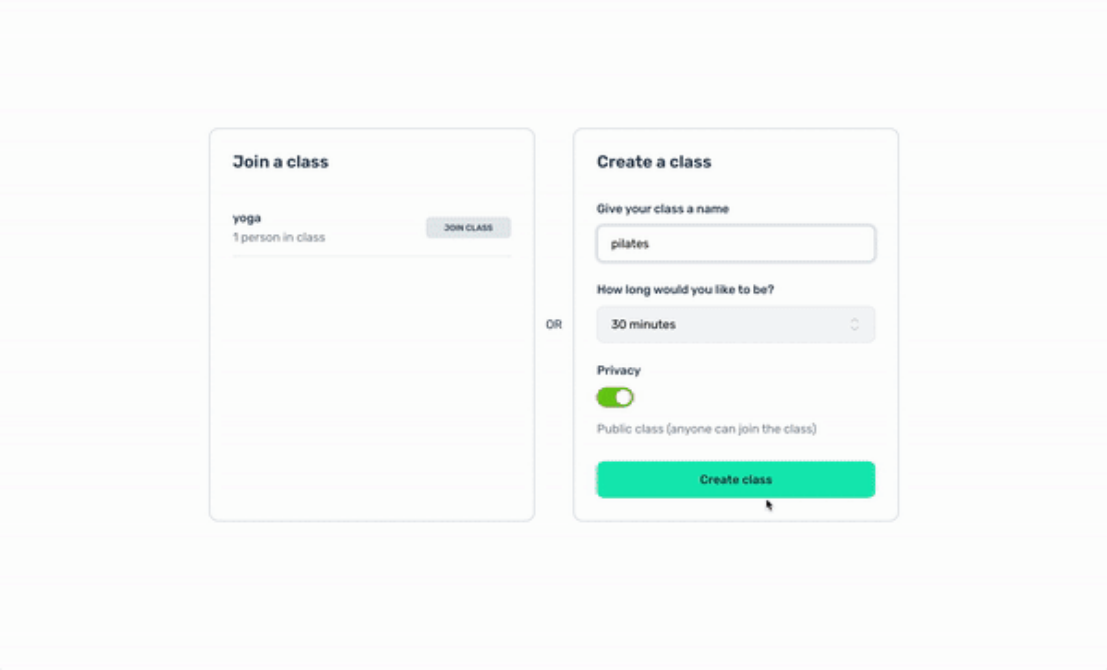
   // Add these to make a cloud (server-side) recording start on
  if (isOwner) {
    properties['enable_recording'] = 'cloud';
    properties['start_cloud_recording'] = true;
  }

  const options = {
    method: 'POST',
    headers: {
      'Content-Type': 'application/json',
      Authorization: `Bearer ${process.env.DAILY_API_KEY}`,
    },
    body: JSON.stringify({ properties }),
  };

  const dailyRes = await fetch(
    `${
      process.env.DAILY_REST_DOMAIN || 'https://api.daily.co/v1'
    }/meeting-tokens`,
    options
  );
```

token.js

Now, when a new class is created and joined, the recording will start automatically and your class instructor can focus on getting started!



Auto-recording enabled when the instructor joins the fitness demo

If you're looking closely at these gifs, you might have also noticed the default recording type for this demo (`local`) is updated to `cloud` too, as shown in the recording modal. 😎

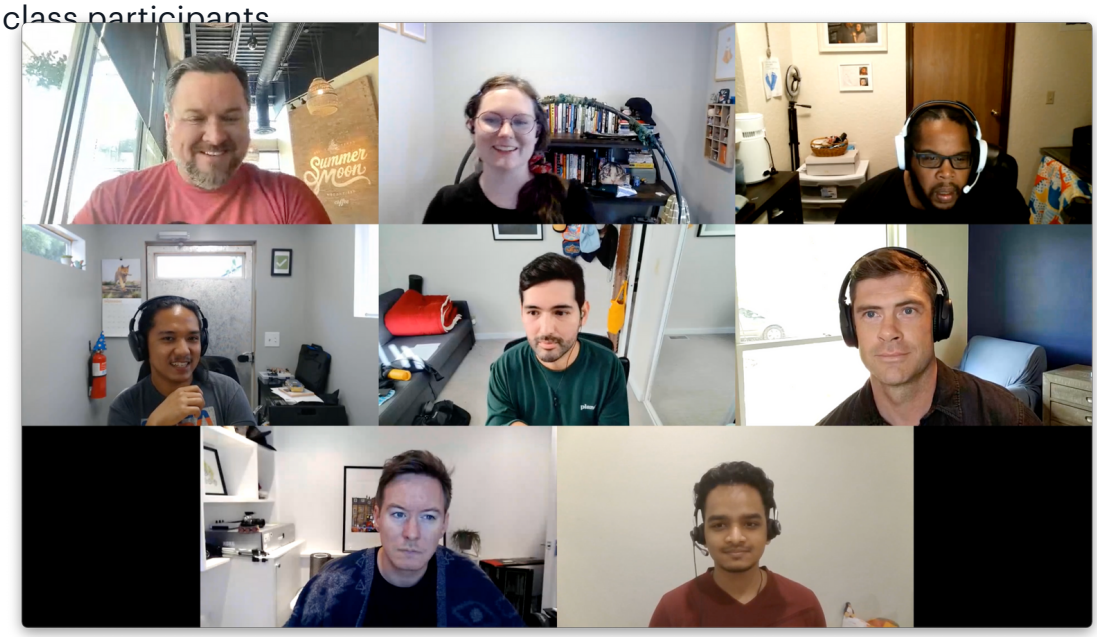
So, now that we've made our fitness class instructor's life a little easier, how can we help out the class participants? One of the easiest ways to encourage folks to loosen up and make your recordings more re-usable is to only record the instructor. This means you can share the recording later for anyone who prefers pre-recorded classes while still maintaining your live class attendees' privacy.

Let's see how to do this.

Updating cloud recording settings to record a single participant in a group call

In our simple gif example above of the app UI, there's only one person in the call so there would only be one person in the recording. If there was

a whole class of participants, our recording would include all of the class participants.



Grid (default) view for group call recordings

Here's an example of a fitness demo recording with a few people joining the class:



Grid view for a fitness demo recording

This isn't ideal when you really only need the instructor recorded, though. Thankfully, we can update the recording layout for cloud recordings. There are several options for the layout but in this case we want to record a single participant – the meeting owner, i.e. the class instructor.

💡 ***Checkout our recording guide for more information on cloud recording layouts.***

To do this, we can use the `updateRecording` method to change the layout type right after the recording starts.

```
await callObject.updateRecording({
  layout: {
    preset: 'single-participant',
    session_id: 'instructor-abcd123...',
  },
});
```

In this code snippet, we're updating the layout `preset` to `'single-participant'` and passing the class instructor's participant `session_id` so `daily-js` knows which single participant should be recorded.

This code snippet can be used in any `daily-js` app and where you put it will depend on the structure of your specific app. As an example, let's take a look at where we could put it in our fitness demo app.

Updating our fitness demo to use single participant recording

In our fitness app, we use a `RecordingProvider` component to manage all Daily recording events. You don't have to set up your app this way – you don't even have to use React! – but let's see how to

update our `RecordingProvider` file to use the single participant cloud layout.

`daily-js` is event-based, which means any time something happens in the call, an event is sent to call participants in the client. When a recording is started, the `'recording-started'` event is sent to all call participants.

We can use this event to know when the recording has started and automatically update the recording layout.

```
import { useParticipants } from '@custom/shared/contexts/Partici
…
export const RecordingProvider = ({ children }) => {
…
 const { participants, isOwner, localParticipant } = useParticip
…
/**
  * Handle recording started.
  */
 const handleRecordingStarted = useCallback(
   (event) => {
     // Added code to update the layout
     if (isOwner) {
       // This is the same code block we reviewed in the last se
       callObject.updateRecording({
         layout: {
           preset: 'single-participant',
           session_id: localParticipant.sessionId,
         },
       });
     }

     …
   },
   [
     recordingState,
     recordingStartedDate,
     callObject,
     isOwner,
     localParticipant,
   ]
 );
```

RecordingProvider.js

Here we check if the local participant is the meeting owner with the `isOwner` variable. If they are, we get the `sessionId` from the `localParticipant` and call `updateRecording` with the local participant's session ID also passed.

Note: This code snippet is not using Daily React Hooks but you could also get these values using the `useLocalParticipant` hook. We are in the process of converting this repo to use Daily React Hooks at the time of writing this and recommend using it if you're writing your own Daily code! 🚀

Assuming this update is successful, the recording will now only record the meeting owner's video and audio and the class participants can enjoy the class with a little more privacy.

Here's an example of a recording with multiple people in the class but only the instructor being recorded:



Single participant recording view in a group call

Wrapping up

And there you have it! We've gone from having a fitness class app that allows for local recordings triggered by a button click to having:

- Cloud (server-side) recordings started as soon as an instructor joins
- An updated recording layout so only the instructor is recorded

To learn more about recordings, check out the Daily recordings guide. You may also enjoy this previous tutorial on building custom recording UI for your app.

Get updates from Daily

Don't miss a thing! Subscribe to our monthly newsletter and we'll keep you up-to-date with the latest product releases, tutorials, and more from Daily.

Email*

Country*

Join peerConnection - our video/WebRTC developer community.

By clicking submit below, you consent to allow Daily.co to store and process the personal information submitted above to provide you the content requested.

Submit

More in **Fitness use case**

Fitness app use case: Adding an “incognito mode” for participants

2 May 2022 - 8 min read

Mute fitness class participants in a video chat app when the instructor joins

24 Mar 2022 - 6 min read

Daily use cases: Building a real-time video fitness class app with Daily

10 Mar 2022 - 4 min read

See all 3 posts →



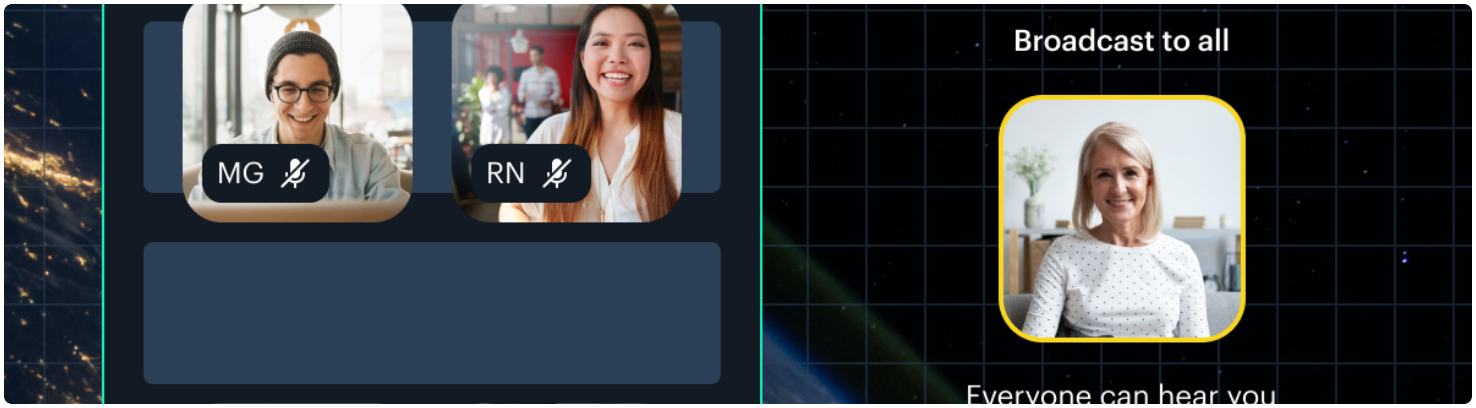


Building e-learning applications with Daily

Showcasing key e-learning features with our virtual class demo



Liza Shulyayeva
4 Apr 2022 – 10 min read



Spatialization

Testing our spatialization features with robots and Jest (Part 7)

Manual and automated testing of spatial video and spatial audio



Liza Shulyayeva
28 Mar 2022 – 14 min read