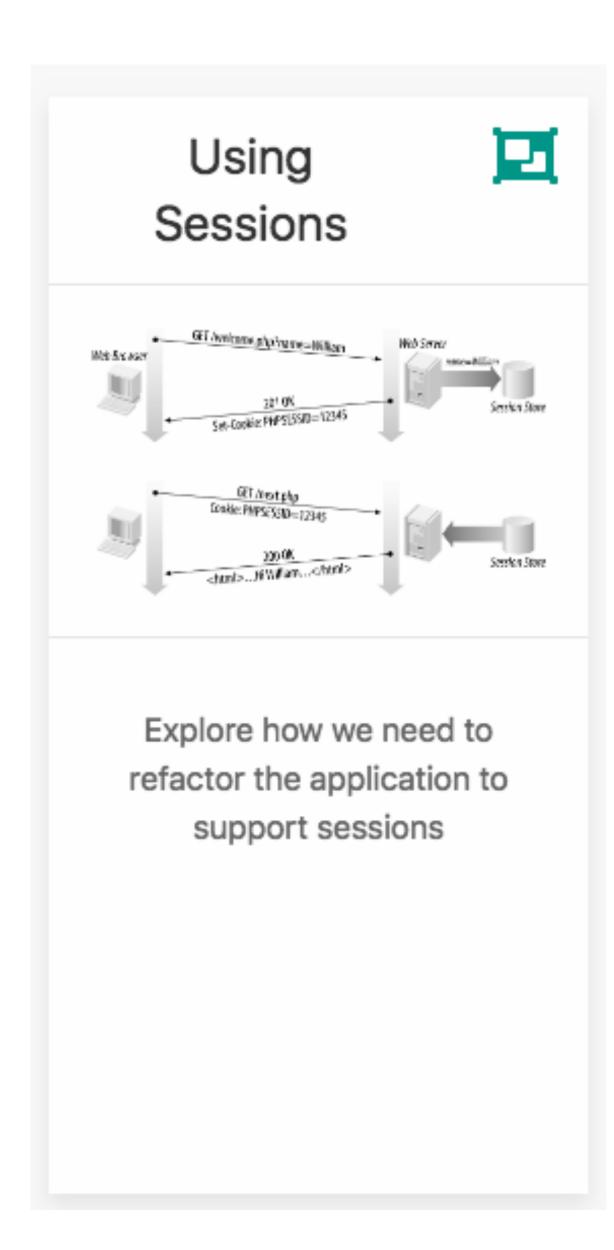
Using Sessions



Playlist Model

```
"playlistCollection": [
    "id": "b76204cc-515a-46ff-b1dc-4efecb8d6ae2",
    "title": "Beethoven Sonatas",
    "songs":
        "id": "072d6417-6d26-4a78-a562-d3f1b78f4313",
        "title": "Sonata No 1",
        "artist": "Beethoven"
        "id": "194d0ecd-dd62-4d6c-83a9-096275026455",
        "title": "Sonata No 2",
        "artist": "Beethoven"
```

Playlist-V3 (no user support)

Revised Playlist Model

```
"playlistCollection": [
    "id": "b76204cc-515a-46ff-b1dc-4efecb8d6ae2",
    "userid": "2b6f0989-7b7f-4a38-ad26-aa06b922d751",
    "title": "Beethoven Sonatas",
    "songs":
        "id": "072d6417-6d26-4a78-a562-d3f1b78f4313",
        "title": "Sonata No 1",
        "artist": "Beethoven"
        "id": "194d0ecd-dd62-4d6c-83a9-096275026455",
        "title": "Sonata No 2",
        "artist": "Beethoven"
```

Revised Playlist Model

```
"playlistCollection": [
additional
                       "id": "b76204cc-515a-46ff-b1dc-4efecb8d6ae2",
property
                       "userid": "2b6f0989-7b7f-4a38-ad26-aa06b922d751",
                       "title": "Beethoven Sonatas",
identifies
                       "songs": [
 user id
                           "id": "072d6417-6d26-4a78-a562-d3f1b78f4313",
                           "title": "Sonata No 1",
                           "artist": "Beethoven"
                           "id": "194d0ecd-dd62-4d6c-83a9-096275026455",
                           "title": "Sonata No 2",
                           "artist": "Beethoven"
```

Playlist belongs to specific user

playliststore

```
New method to
    retrieve all playlists
                                       Id of user passed as
  belonging to a specific
                                             parameter
            user
const playlistStore = {
 getUserPlaylists(userid) {
   return this.store.findBy(this.collection, { userid: userid });
 },
```

dashboard: index

Discover which user is currently logged in

```
const dashboard = {
  index(request, response) {
    logger.info('dashboard rendering');
    const loggedInUser = accounts.getCurrentUser(request);
    const viewData = {
      title: 'Playlist Dashboard'
      playlists: | playlistStore.getUserPlaylists(loggedInUser.id),
    };
    logger.info('about to render', playlistStore.getAllPlaylists());
    response render ('dashboard', viewData);
  },
```

Retrieve only those playlists associated with the logged in user

dashboard: addPlaylist

Discover which user is currently logged in

```
const dashboard = {
addPlaylist(request, response) {
  const loggedInUser = accounts.getCurrentUser(request);
  const newPlayList = {
    id: uuid(),
    userid: loggedInUser.id,
    title: request.body.title,
    songs: [],
  };
logger.debug('Creating a new Playlist', newPlayList);
  playlistStore.addPlaylist(newPlayList);
  response.redirect('/dashboard');
},
                                        In the new playlist,
                                        include the id of the
                                     currently logged in user
```