

Quickswitch

←

→

↺

↻

Vulnerable Code

Jump

<

>

*gradebook

*api

dev

test

__init__.py

admin.py

apps.py

models.py

pagination.py

permissions.py

serializers.py

tests.py

urls.py

*views.py

app

comments

grades

users

manage.py

.gitignore

requirements.txt

212

API endpoint for viewing public courses.

213

"""

214

queryset = Course.objects.filter(is_active=True).order_by('start_date')

215

serializer_class = CoursePublicSerializer

216

permission_classes = (AllowAny,)

217

218

def get_object(self):

219

"""

220

Returns the object the view is displaying.

221

"""

222

queryset = self.filter_queryset(self.get_queryset())

223

224

Perform the lookup filtering.

225

lookup_url_kwarg = self.lookup_url_kwarg or self.lookup_field

226

227

assert lookup_url_kwarg in self.kwargs, (

228

'Expected view %s to be called with a URL keyword argument '

229

'named "%s". Fix your URL conf, or set the `.lookup_field` '

230

'attribute on the view correctly.' %

231

(self.__class__.__name__, lookup_url_kwarg)

232

)

233

234

filter_kwargs = {self.lookup_field: self.kwargs[lookup_url_kwarg]}

235

results = list(queryset.raw(

236

'SELECT * FROM grades_course WHERE id=%s' % filter_kwargs.get('pk')))

237

238

if not results:

239

raise NotFound

240

241

obj = results.pop()

242

243

May raise a permission denied

244

self.check_object_permissions(self.request, obj)

245

246

return obj

247

248

def retrieve(self, request, *args, **kwargs):

249

instance = self.get_object()

250

serializer = self.get_serializer(instance)

251

return Response(serializer.data)

252

253

254

class CourseViewSet(viewsets.ReadOnlyModelViewSet):

212

213

course.objects.filter(is_active=True).order_by('start_date')

214

class = CoursePublicSerializer

215

asses = (AllowAny,)

216

217

def get_queryset(self):

218

"""

219

Returns the object the view is displaying.

220

"""

221

queryset = self.filter_queryset(self.get_queryset())

222

223

Perform the lookup filtering.

224

lookup_url_kwarg = self.lookup_url_kwarg or self.lookup_field

225

226

assert lookup_url_kwarg in self.kwargs, (

227

'Expected view %s to be called with a URL keyword argument '

228

'named "%s". Fix your URL conf, or set the `.lookup_field` '

229

'attribute on the view correctly.' %

230

(self.__class__.__name__, lookup_url_kwarg)

231

)

232

233

filter_kwargs = {self.lookup_field: self.kwargs[lookup_url_kwarg]}

234

obj = generics.get_object_or_404(queryset, **filter_kwargs)

235

236

May raise a permission denied

237

self.check_object_permissions(self.request, obj)

238

239

return obj

240

241

def retrieve(self, request, *args, **kwargs):

242

instance = self.get_object()

243

serializer = self.get_serializer(instance)

244

return Response(serializer.data)

245

246

247

class CourseViewSet(viewsets.ReadOnlyModelViewSet):

248

"""

249

API endpoint for viewing courses.

250

"""

251

serializer_class = CoursePublicSerializer

252

permission_classes = (IsAuthenticated,)

253

254

def get_queryset(self):

Solution 1

Solution 2

Solution 3

Solution 4

Accept

Hint

⚙

Licenses © Secure Code Warriors 2021