

Projeto escolhido: Django

O Django fornece vários decoradores que podem ser aplicados a visualizações para suportar vários recursos HTTP.

Os decoradores em `django.views.decorators.http` podem ser usados para restringir o acesso às visualizações com base no método de solicitação. Esses decoradores retornarão um `django.http.HttpResponseNotAllowed` se as condições não forem atendidas.

Os decoradores a seguir `django.views.decorators.http` podem ser usados para controlar o comportamento do cache em visualizações específicas.

```
condition( etag_func = Nenhum , last_modified_func = Nenhum )
etag( etag_func )
last_modified( last_modified_func )
```

Esses decoradores podem ser usados para gerar cabeçalhos ETag Last-Modified; Os clientes HTTP podem enviar vários cabeçalhos para informar ao servidor sobre as cópias de um recurso que eles já viram. Isso é comumente usado ao recuperar uma página da Web (usando uma GET request HTTP) para evitar o envio de todos os dados de algo que o cliente já recuperou. No entanto, as mesmas headers pode ser utilizado para todos os métodos de HTTP (POST, PUT, DELETE, etc.).

Para cada página (resposta) que o Django envia de volta de uma visão, ele pode fornecer dois cabeçalhos HTTP: o ETagheader e o Last-Modified cabeçalho . Esses cabeçalhos são opcionais em respostas HTTP. Eles podem ser definidos por sua função de visualização ou você pode contar com o `ConditionalGetMiddleware` para definir o ETag header.