Contents

1 Development 3
an app that adds user reminders for the various activities that you, the site builder would like to guide users to doing and completing.
The source repository can be found at https://github.com/eldarion/django-reminders

Contents

ChangeLog

0.2

• added permanent dismissals

Migrations

Here is a sample migration that should work with Postgresql/nashvegas:

```sql
### New Model: reminders.Dismissal
CREATE TABLE "reminders_dismissal" (
   "id" serial NOT NULL PRIMARY KEY,
   "user_id" integer NOT NULL REFERENCES "auth_user" ("id") DEFERRABLE INITIALLY DEFERRED,
   "label" varchar(200) NOT NULL,
   "dismissed_at" timestamp with time zone NOT NULL
);
CREATE INDEX "reminders_dismissal_user_id" ON "reminders_dismissal" ("user_id"");
```

0.1

• initial release
Installation

- To install django-reminders:

```
pip install django-reminders
```

- Add 'reminders' to your INSTALLED_APPS setting:

```
INSTALLED_APPS = (  
    # other apps
    "reminders",
)
```

- Finally (and optionally if you configure all your reminders to not be dismissable):

```
...  
url(r'^reminders/', include("reminders.urls")),  
...
```

Template Tags

reminders

The reminders tag loops through the REMINDERS list of callables and evaluates them for the current user (requires the request object to be in context. If the callable returns a dict instead None, then it will evaluate the message in the same tuple and add that to the results list. After evaluating all callables it sets a context variable to the list:

```
{% reminders as user_reminders %}
```

Settings

REMINDERS

This app is driven by a list of callables and associated messages configured by this setting. Here is an example:

```
from emailconfirmation.reminders import confirmed
REMINDERS = {  
    "profile_completed": {  
        "test": "profiles.reminders.completed",  
        "message": "You have only completed $(percentage)s%% of your <a href="%(url)s→">profile</a>.",  
        "dismissable": "permanent"  
    },  
    "email_confirmed": {  
        "test": lambda user: confirmed(user),  
        "message": "Please <a href="%(url)s">confirm</a> your email address.",  
        "dismissable": "no"  
    }  
}
```

Valid values for the dismissable key are permanent, session, and no. If left out of the settings it will default to session. As you might have guessed, this controls whether or not a user can dismiss a reminder and if they do, whether it is dismissed for the duration of their session or for good.
Callable API

This callables may be provided by third-party apps or may be defined by you, the site developer. In either case, they should follow the following conventions:

```python
def name(user):
    if there_is_stuff_for(user):
        return build_up_dict_for(user)
```

The message in the tuple with this callable will need to know what data is being supplied by the callable. If there isn’t a reminder, then the callable should return None.

Usage

After configuring the appropriate settings, you will need to implement any of the callables that you listed in the REMINDERS setting. Once that has been done, using django-reminders is as simple as using a single template tag.

Example:

```python
{% load reminders_tags %}
<h3>Reminders</h3>
{% reminders as user_reminders %}
{% if user_reminders %}
<ul>
    {% for reminder in user_reminders %}
    <li>
        {{ reminder.message }}
        {% if reminder.dismiss_url %}
        <a href="{{ reminder.dismiss_url }}">Dismiss</a>
        {% endif %}
    </li>
    {% endfor %}
</ul>
{% else %}
<p class="info">You have no reminders at this time.</p>
{% endif %}

You’ll want to hook up the dismiss link to an AJAX post as that URL will only response to POST methods.