1 Indices and tables 1
Indices and tables

• genindex
• modindex
• search

\textbf{class emailviews.views.SendEmailMixin}

Base mixin for sending emails. Uses `django.core.mail.send_email` to send the email. By default, to construct an email we will use `django.template.loader.render_to_string` passing `email_subject_template` and `email_body_template` as paths to the templates.

\textbf{class emailviews.views.ActivationEmailViewMixin}

Provides methods for generating and validating activation_keys. Uses `django.core.signing.TimestampSigner` allowing activation keys to expire after certain period of time.

\textbf{You have to implement 2 methods} while subclassing this view:

\textbf{emailviews.views.ActivationEmailViewMixin.get_salt}

\textbf{emailviews.views.ActivationEmailViewMixin.get_max_age}

\textbf{generate_activation_key (value: str) \rightarrow str}

Generates activation key via `django.core.signing.dumps`.

\textbf{get_max_age ()}

Has to return time expressed in seconds after which activation_key expires.

\textbf{get_request_root_url ()}

Generates root url of your website based on request. Example that you can use in your template:

\begin{verbatim}
{{root_url}}{% url 'activation' activation_key=activation_key %}
\end{verbatim}

\textbf{get_salt ()}

Has to return secret_key used to sign and validate values.

\textbf{validate_activation_key (key: str) \rightarrow str}

Validates key using `django.core.signing.loads` If value is not validated (either time expires, or key is wrong) returns None.
A
ActivationEmailViewMixin (class in emailviews.views), 1

G
generate_activation_key() (emailviews.views.ActivationEmailViewMixin method), 1
get_max_age() (emailviews.views.ActivationEmailViewMixin method), 1
get_request_root_url() (emailviews.views.ActivationEmailViewMixin method), 1
get_salt() (emailviews.views.ActivationEmailViewMixin method), 1

S
SendEmailMixin (class in emailviews.views), 1

V
validate_activation_key() (emailviews.views.ActivationEmailViewMixin method), 1