

In Chapter 10, we will add account activation and password reset features to our sample Rails application. They'll be created as a separate resources and add useful features for the users of our application.

- 10.1 Account activation
  - 10.1.1 Account activations resource
    - Create the account activations resource.
    - Modification of routes to include 'account activations' resource with an 'edit' action.

```
resources :account_activations, only: [:edit]
```

- Activation tokens to activate users. Tokens are objects that represent something else.
- Seeds are used to populate the database with data.
- Fixtures are a way of organizing data that you want to test against; in short, it is sample data.

```
michael:  
  name: Michael Example  
  email: michael@example.com
```

- 10.1.2 Account activation mailer method
  - Action Mailer allows you to send emails from your application using mailer classes and views.
  - Create a new mailer. Two templates:
    - Plain-text
    - HTML
  - Mailer class

```
class ApplicationMailer < ActionMailer::Base  
  default from: "noreply@example.com"  
  mail to: user.email, subject: "Account activation"  
  attachments['filename.jpg'] = File.read('/path/to/filename.jpg')  
  layout 'awesome' # use awesome.(html|text).erb as the layout  
end
```

- 10.1.3 Activating the account
  - `user = User.find_by(email: params[:email])`  
`if user && user.authenticated?(:activation, params[:id])`
  - Add a check on the user model to determine if the token provided matches the hash value or digest. Uses Bcrypt library.
  - Create a method on the session helper to return the current user.
  - Add AccountActivationsController to handle activation logic

- Add activation logic to session controller to ensure that each user who logs in has been successfully activated.
  - 10.2 Password reset
    - 10.2.1 Password resets resource
      - Create a PasswordResets Controller

```
resources :password_resets, only: [:new, :create,
:edit, :update]
```

- Update our view to contain a link to this resource

```
<%= link_to "(forgot password)",
new_password_reset_path %>
```

- Add reset metadata to the users table

- 10.2.2 Password resets controller and form

- `<%= form_for(:password_reset, url: password_resets_path) do |f| %>`

The form\_for method yields a form builder object (the `f` variable). Methods to create form controls are called on the form builder object `f`.

- Before Filters are used to specify actions that take place prior to another action. They help adhere to business rules and can automate repetitive procedures.

```
before_save :downcase_email

# Converts email to all lower-case.
def downcase_email
  self.email = email.downcase
end
```

- 10.2.3 Password reset mailer method

- `UserMailer.password_reset(self).deliver_now`